18 JUEGOS DINAMICOS PARA TU MSX

P. MONSAUT

EDITORIAL NORAY

18 Juegos dinámicos para tu MSX

P. MONSAUT

EDITORIAL NORAY, S.A.
San Gervasio de Cassolas, 79
Tel. 211 11 46 - 08022 Barcelona

Cualquier duda o aclaración sobre esta obra, será contestada por el departamento técnico de Editorial Noray, siempre y cuando se solicite por escrito al Apartado de correos n.º 6015, 08080 de Barcelona.

Título original: MSX Jeux d'action
Traducción de: Manuel Figueras

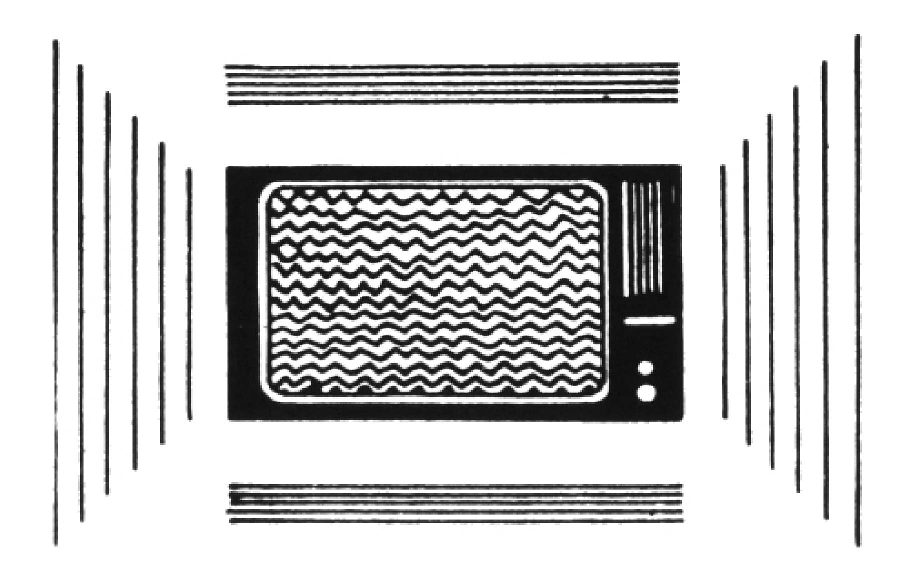
Primera edición, 1985
Depósito Legal: B.36367-1985
ISBN: 84-7486-053-9
Número de edición de E.N., 88
Printed in Spain - Impreso en España
Gráficas Instar - C. Industria, s/n - Hospitalet de Llobregat

Indice

Prólogo

Los juegos presentados en este libro utilizan funciones propias de los microordenadores MSX: sonido, color, gráficos animados, etc. Por dicho motivo, estos programas sólo sirven para aparatos de este standard y difícilmente pueden adaptarse a otros productos.

Aparte del propio juego, el estudio de estos programas le va a enseñar numerosas técnicas de programación que le serán muy útiles para desarrollar sus propios programas.



NOTAS CONCERNIENTES A LOS PROGRAMAS

Los programas han de ser tecleados tal como se reproducen en casette o en diskette al objeto de evitar tener que volver a teclearlos cada vez que se utilizan. Para ello emplee la instrucción

C SAVE "<nombre>"

sustituyendo nombre por el nombre del programa a grabar.

Ejemplo: Para grabar el programa BOMBARDEO AEREO, pulsar

C SAVE "BOMBARDEO AEREO"

y apretar la tecla ENTER una vez apretadas las teclas PLAY y RECORD del magnetófono. Para recargar un programa hay que rebobinar la casette hasta el inicio del programa y pulsar CLOAD "<nombre>". Por ejemplo, si quiere recargar el programa BOMBARDEO AEREO, hay que pulsar

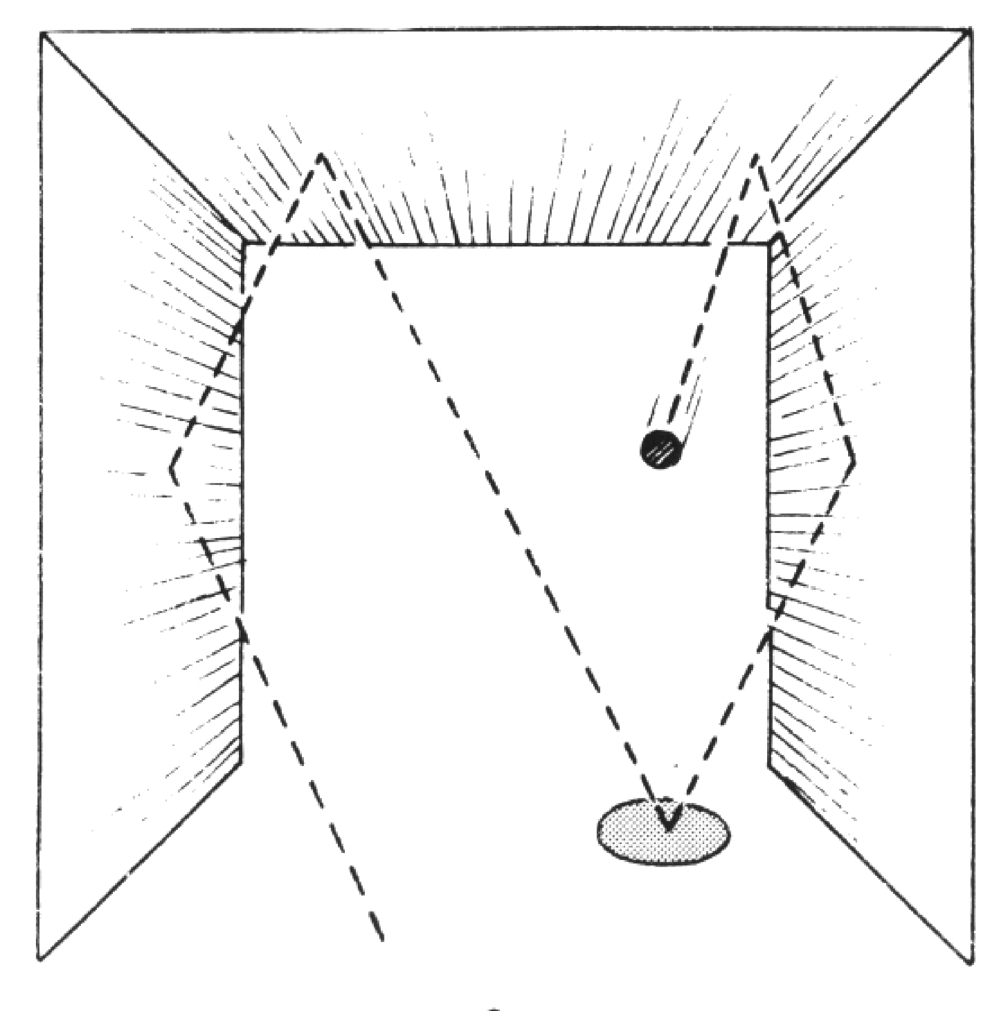
CLOAD "BOMBARDEO AEREO"

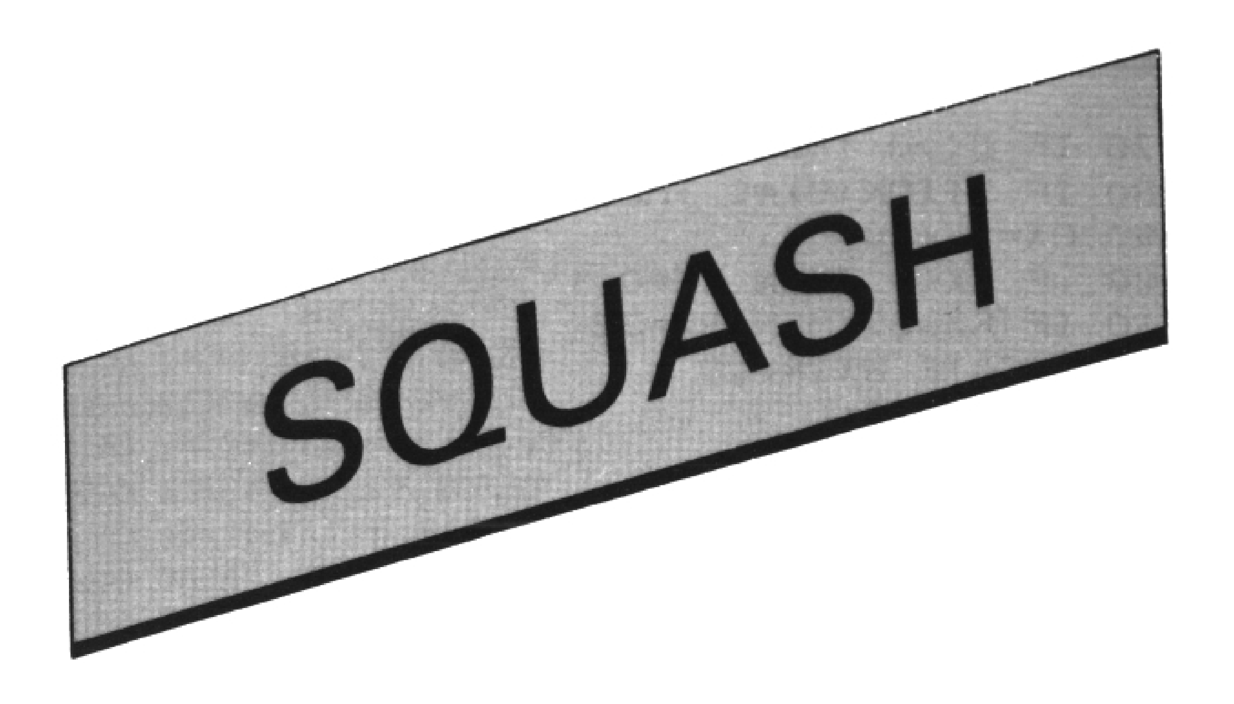
y apretar la tecla ENTER.

Al objeto de reducir los errores al máximo, los listados de los programas han sido reproducidos fotográficamente. Si sus programas no funcionan, hay que verificarlos atentamente, vigilando de modo especial los puntos siguientes:

— líneas omitidas
— números de líneas equivocados
— errores de pulsación (mucha atención para no confundir la letra O y el número 0, o la letra I y el número 1).

Todos los programas que utilizan las teclas del cursor pueden utilizar un joystick. Para ello hay que reemplazar STICK(0) por STICK(1) y STRIG(0) por STRIG(1).



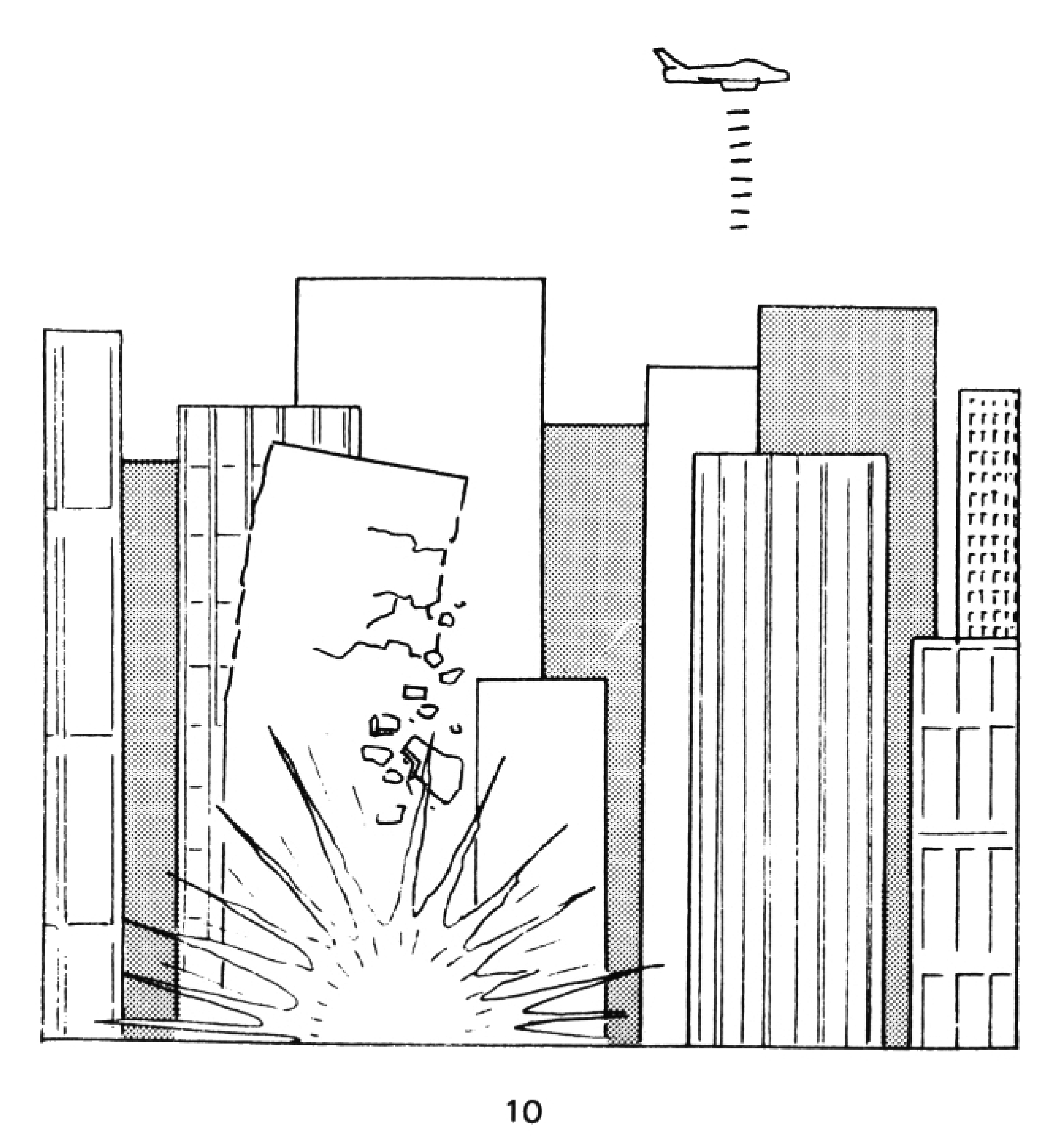


Gracias a su ordenador, puede jugar a squash cómodamente sentado en un sillón. La raqueta se desplaza con ayuda de las teclas de control del cursor. Dispone de diez pelotas que ha de mantener en juego el mayor tiempo posible. Cada pelota que se devuelve proporciona un punto.

```
10 REM **********
20 REM * SQUASH *
30 REM **********
40 KEY OFF
50 GOSUB 540
60 LOCATE BX,BY,0
70 PRINT N$
80 BX=BX+DX
90 BY=BY+DY
100 LOCATE BX,BY,0
110 PRINT B$;
120 IF BY=21 AND ABS(BX-RX-3)>1 THEN 250
130 IF BY=21 THEN S=S+1:BEEP:DY=-DY
140 IF BY=1 THEN BEEP:DY=-DY
150 IF BX=3 OR BX=35 THEN BEEP:DX=-DX
160 D=2*((STICK(0)=7)-(STICK(0)=3))
```

```
170 IF D<>0 THEN DO=D
180 IF STICK(0)=0 THEN DO=0
190 RX=RX+DO
200 IF RX<0 THEN RX=0
210 IF RX>32THEN RX=32
220 LOCATE RX,RY,0
230 PRINT R$;
240 GOTO 60
250 NB=NB+1
260 IF NB=11 THEN 370
270 LOCATE BX,BY,0
280 PRINT N$;
290 FOR I=1 TO 3
300 BEEP
310 FOR J=1 TO 100
320 NEXT J
330 NEXT I
340 DO=0
350 GOSUB 720
360 GOTO 60
370 LOCATE 13,5,0
380 PRINT "PUNTUACION :";S;
390 IF S>R1 THEN R1=S
400 LOCATE 13,10,0
410 PRINT "RECORD :";R1;
420 LOCATE 13,15
430 PRINT "OTRA ?";
440 NB=0
450 S=0
460 D$=INKEY$
470 IF D$<>"" THEN 460
480 D$=INKEY$
490 IF D$="" THEN 480
500 IF D$<>"N" AND D$<>"n" THEN 50
510 CLS
520 LOCATE 0,0,1
530 END
540 CLS
550 SCREEN 0,0
560 COLOR 15,9
570 DEFINT A-Z
```

```
580 N$=CHR$(32)
590 FOR BX=2 TO 37
600 LOCATE BX,0,0
610 PRINT CHR$(219);
620 NEXT BX
630 FOR BY=1 TO 21
640 LOCATE 2,BY,0
650 PRINT CHR$(219);
660 LOCATE 37,BY,0
670 PRINT CHR$(219);
680 NEXT BY
690 R$=N$+N$+CHR$(223)+CHR$(223)+CHR$(22
3)+N$+N$
700 B$=CHR$(249)
710 RX=16
720 RY=22
730 BY=21
740 BX=INT(RND(1)*32)+5
750 DY=-1
760 DX=(INT(RND(1)*2)-.5)*2
770 RETURN
```





Su misión consiste en destruir la ciudad que está sobrevolando, con objeto de poder aterrizar. A cada pasada, su avión vuela un poco más bajo. No puede lanzar una bomba (pulsando sobre una tecla cualquiera) hasta que la bomba precedente haya alcanzado su objetivo o el suelo. Cuando su avión ha aterrizado (o cuando se ha estrellado contra un edificio), se registra la puntuación, así como el récord del día. Si este juego le parece demasiado difícil, puede variar los límites de la ciudad (10 y 30, línea 730).

```
10 REM ********************
20 REM * BOMBARDEO AEREO *
30 REM ********************
40 KEY OFF
50 WIDTH 39
60 GOSUB 590
70 GOSUB 670
80 FOR HY=0 TO 23
90 FOR HX=0 TO 38
100 LOCATE HX,HY,0
110 PRINT A$;
120 IF HX>35 THEN 140
130 IF VPEEK(HX+5+HY*40)<>32 THEN 240
140 IF INKEY$<>"" AND BY=0 AND HX<>38 TH
EN BX=HX+1:BY=HY+1
150 IF BY<>0 THEN GOSUB 510 ELSE GOSUB 5
60
```

```
160 NEXT HX
170 LOCATE 37,HY,0
180 PRINT NN$;
190 NEXT HY
200 FOR I=1 TO 500
210 NEXT I
220 S=S+40*HY+HX
230 GOTO 70
240 S=S+40*HY+HX
250 VPOKE BX+B1*40,0
260 FOR I=HY TO 22
270 LOCATE HX+1,I,0
280 PRINT NN$;
290 LOCATE HX+1,I+1,0
300 PRINT RIGHT$(A$,3);
310 FOR J=1 TO 100
320 NEXT J
330 NEXT I
340 IF S>R THEN R=S
350 LOCATE 3,0,0
360 PRINT "PUNTUACION :";S;
370 LOCATE 25,0,0
380 PRINT "RECORD :";R;
390 FOR I=1 TO 100
400 NEXT I
410 R$=INKEY$
420 IF R$<>"" THEN 410
430 LOCATE 7,13,0
440 PRINT "OTRA ?";
450 R$=INKEY$
460 IF R$="" THEN 450
470 IF R$<>"N" AND R$<>"n" THEN S=0:GOTO
 70
480 COLOR 15,4,5:LOCATE 0,0,1
490 CLS
500 END
510 IF BY>23 THEN BY=0
520 VPOKE BX+B1*40,32
530 B1=BY
540 IF BY<>0 THEN VPOKE BX+BY*40,B:BY=BY
+1
```

```
550 RETURN
560 FOR I=1 TO 10
570 NEXT I
580 RETURN
590 SCREEN 0,0
600 GOSUB 790
610 COLOR 10,1
620 DEFINT A-V
630 A$=CHR$(32)+CHR$(128)+CHR$(129)+CHR$
(130)
640 B=131
650 NN$=CHR$(32)+CHR$(32)+CHR$(32)
660 RETURN
670 HX=0
680 HY=0
690 BX=0
700 BY=0
710 B1=BY
720 CLS
730 FOR I=15 TO 29
740 FOR J=23 TO INT(RND(1)*5)+14 STEP-1
750 VPOKE I+J*40,132
760 NEXT J
770 NEXT I
780 RETURN
790 RESTORE
800 FOR I=0 TO 39
810 READ A
820 VPOKE 3072+I,A*4
830 NEXT I
840 RETURN
850 DATA 0,0,32,48,56,63,63,63
860 DATA 0,0,0,0,0,63,63,63
870 DATA 0,0,0,0,0,48,60,62
880 DATA 14,14,4,14,14,14,14,4
890 DATA 63,41,41,41,63,63,63,63
```

BOUM
BOUM
BOUM
BOUM
BOUM

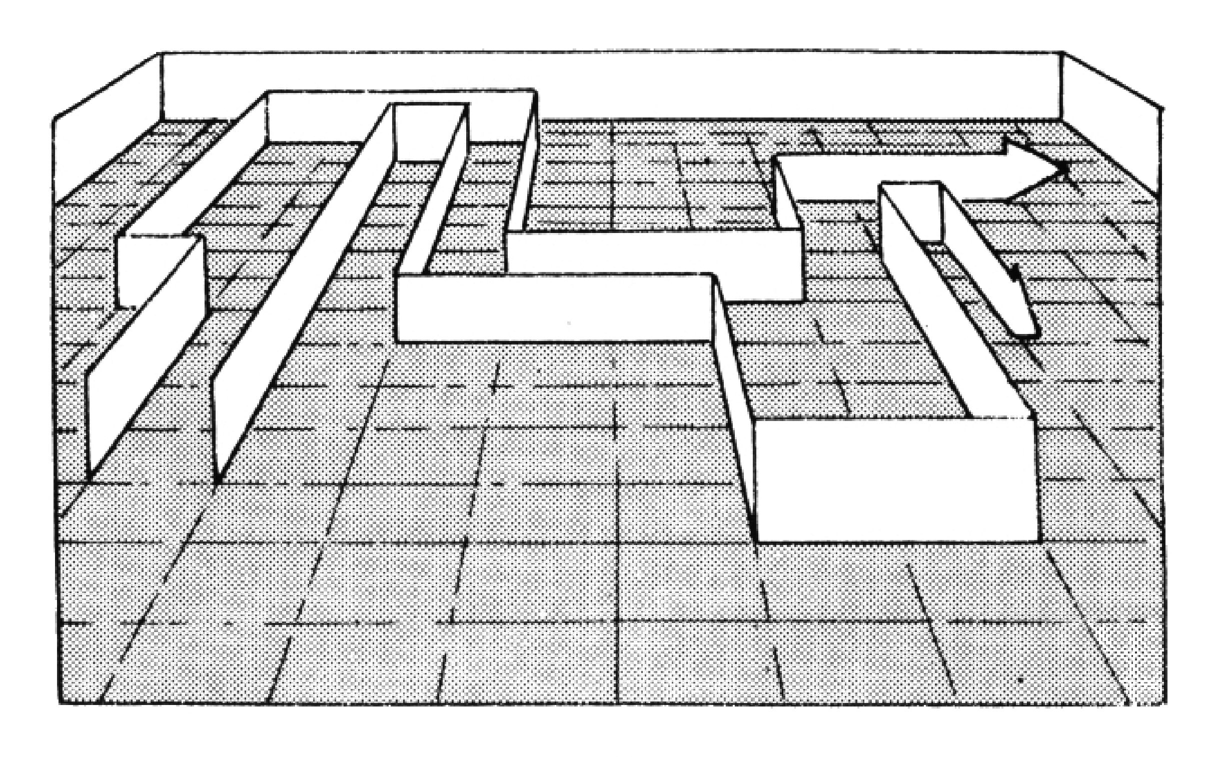


Ahora se han invertido los papeles. Su misión es manejar las defensas contra aviones e intentar abatir los aviones que vuelan sobre su posición. Para disparar hay que usar una tecla cualquiera. Inicialmente dispone de 15 misiles. Si consigue abatir 10 aviones, obtendrá una bonificación de 10 puntos y 10 nuevos misiles adicionales.

```
10 REM ***********
20 REM * D.C.A. *
30 REM ***********
40 KEY OFF
50 WIDTH 39
60 CLEAR 250
70 GOSUB 640
80 A$=RIGHT$(A$,1)+LEFT$(A$,38)
90 B$=RIGHT$(B$,38)+LEFT$(B$,1)
100 LOCATE 0,0,0
110 PRINT A$;
120 LOCATE 0,2
130 PRINT B$;
140 LOCATE MX-2,24,0
150 PRINT J$;
160 R$=INKEY$
```

```
170 IF R$<>"" AND MY=24 THEN MY=23:CM=G4
:NM=NM-1
180 IF MY<>24 THEN MY=MY-2:VPOKE MX+MY*4
0,CM:VPOKE MX+(MY+2)*40,CO
190 IF MY<>3 THEN 270
200 IF VPEEK(MX+80)=CO THEN 270
210 VPOKE MX+80,CE
220 VPOKE MX+MY*40,CO
230 BEEP
240 S=S+1
250 B$=LEFT$(B$,17)+O1$+RIGHT$(B$,18)
260 GOTO 340
270 IF MY<>1 THEN 370
280 IF VPEEK(MX)=CO THEN 370
290 VPOKE MX,CE
300 S=S+1
310 VPOKE MX+MY*40,CO
320 BEEP
330 A$=LEFT$(A$,17)+O1$+RIGHT$(A$,18)
340 IF S>1 AND S/10=INT(S/10) THEN GOSUB
 570
350 MY=24
360 M=G6
370 LOCATE 0,24,0
380 PRINT "S :";S;"M :";NM;
390 IF NM<1 AND MY=24 THEN 420
400 IF MY<=1 THEN VPOKE MX+40,CO:MY=24:M
=G6
410 GOTO 80
420 IF S>R THEN R=S
430 LOCATE 15,10
440 PRINT "PUNTUACION :";S;
450 LOCATE 15,13
460 PRINT "RECORD :";R;
470 LOCATE 15,16
480 PRINT "OTRA ?";
490 R$=INKEY$
500 IF R$<>"" THEN 490
510 R$=INKEY$
520 IF R$="" THEN 510
530 IF R$<>"N" AND R$<>"n" THEN 70
540 LOCATE 0,0,1
```

```
550 CLS
560 END
570 A$=A1$
580 B$=B1$
590 NM=NM+10
600 FOR I=1 TO 300
610 NEXT I
620 S=S+10
630 RETURN
640 DEFINT A-Z:SCREEN 0,0:COLOR 10,4
650 GOSUB 800
660 S=0:A$="":B$="":N$=CHR$(32)
670 FOR I=1 TO 5
680 A$=A$+CHR$(128)+CHR$(129)+N$+N$+N$+N
$+N$+N$
690 B$=B$+CHR$(130)+CHR$(131)+N$+N$+N$+N
$+N$+N$
700 NEXT I
710 A$=LEFT$(A$,39):B$=LEFT$(B$,39)
720 A1$=A$:B1$=B$:CE=135
730 J$=CHR$(132)+CHR$(133)+CHR$(132)
740 MX=19:MY=24:NM=15:G6=133:G4=134
750 CO=32:O1$=N$+N$+N$+N$
760 LOCATE MX-2,MY,0:PRINT J$;
770 LOCATE 0,0,0:PRINT A$;
780 LOCATE 0,2,0:PRINT B$;
790 RETURN
800 RESTORE
810 FOR I=0 TO 63
820 READ A:VPOKE 3072+I,A*4
830 NEXT I
840 RETURN
850 DATA 0,1,48,49,63,63,16,0
860 DATA 0,62,36,62,63,63,62,24
870 DATA 0,31,9,31,63,63,31,6
880 DATA 0,32,3,35,63,63,2,0
890 DATA 0,0,0,0,0,63,63,63
900 DATA 30,30,30,63,63,63,63,63
910 DATA 4,14,14,14,14,14,14,14
920 DATA 4,21,14,4,4,14,21,4
```





Dos jugadores se enfrentan para dividirse el espacio vital. Cada uno de ellos debe esforzarse, al irse desplazando, para no cortar jamás su trazado o el de su adversario, y sin salirse del rectángulo dibujado en la pantalla. Los mandos a utilizar son los joysticks o las teclas siguientes:

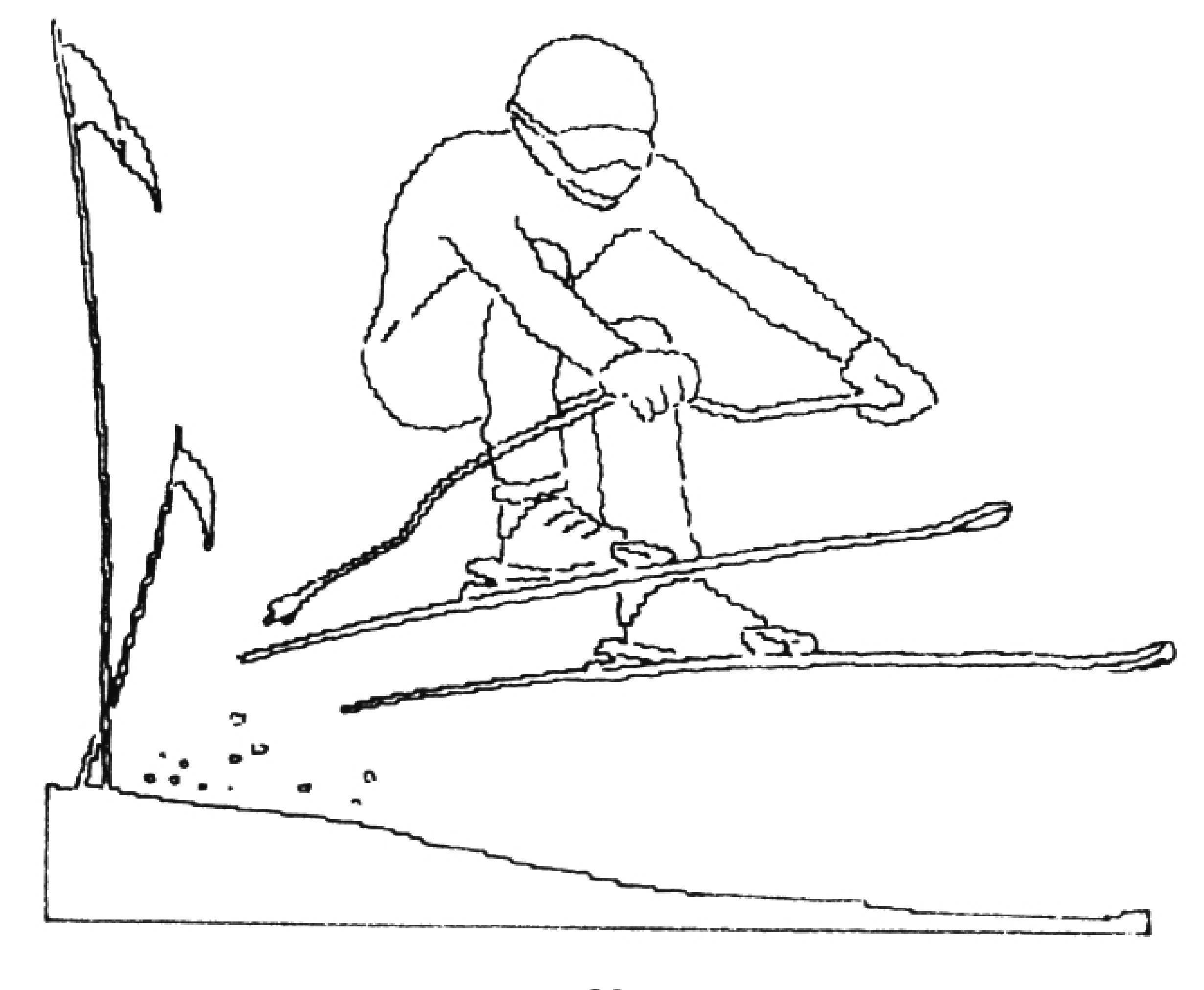
Jugador de la derecha: <P>, <L>, <;> y <.>
Jugador de la izquierda: <W>, <A>, <S> y <Z>

```
10 REM *********
20 REM * TRAZOS *
30 REM *********
40 KEY OFF
50 WIDTH 39
60 GOSUB 690
70 ON JK GOTO 140
80 D$=INKEY$
90 HB=(D$="L")-(D$=";")
100 VB=(D$="P")-(D$=".")
110 HA=(D$="A")-(D$="S")
120 VA=(D$="W")-(D$="Z")
130 GOTO 180
140 HA=(STICK(1)=7)-(STICK(1)=3)
150 VA=(STICK(1)=1)-(STICK(1)=5)
160 HB=(STICK(2)=7)-(STICK(2)=3)
170 VB=(STICK(2)=1)-(STICK(2)=5)
180 IF HA<>0 THEN H1=HA:V1=0
190 IF VA<>0 THEN V1=VA:H1=0
200 IF HB<>0 THEN H2=HB:V2=0
210 IF VB<>0 THEN V2=VB:H2=0
```

```
220 X1=X1+H1
230 Y1=Y1+V1
240 IF VPEEK(X1+Y1*40)<>N THEN 320
250 VPOKE X1+Y1*40,CG
260 X2=X2+H2
270 Y2=Y2+V2
280 IF VPEEK(X2+Y2*40)<>N THEN 380
290 VPOKE X2+Y2*40,CD
300 BEEP
310 GOTO 70
320 F2=F2+1
330 GOSUB 630
340 IF F2=10 THEN 440
350 IF INKEY$<>"" THEN 350
360 GOSUB 770
370 GOTO 70
380 F1=F1+1
390 GOSUB 630
400 IF F1=10 THEN 500
410 IF INKEY$<>"" THEN 410
420 GOSUB 770
430 GOTO 70
440 CLS
450 LOCATE 9,5
460 PRINT "EL JUGADOR DE LA DERECHA HA GANADO ";
470 LOCATE 15,10
480 PRINT F2;"A";F1
490 GOTO 550
500 CLS
510 LOCATE 9,5
520 PRINT "EL JUGADOR DE LA IZQUIERDA HA GANADO";
530 LOCATE 15,10
540 PRINT F1;"A";F2
550 LOCATE 14,15,1
560 PRINT "OTRA  ?";
570 IF INKEY$<>"" THEN 570
580 D$=INKEY$'
590 IF D$="" THEN 580
600 IF D$<>"N" AND D$<>"n" THEN RUN
610 CLS
620 END
630 FOR I=1 TO 5
```

```
640 BEEP
650 FOR J=1 TO 100
660 NEXT J
670 NEXT I
680 RETURN
690 CLS
700 SCREEN 0,0
710 DEFINT A-Z:COLOR 1,12:LOCATE 6,10,0
720 GOSUB 920
730 PRINT "JOYSTICKS ?";
740 R$=INKEY$:IF R$="" THEN 740
750 JK=-(R$="O")-(R$="o")
760 CG=128:CD=129:N=32
770 CLS
780 FOR X1=0 TO 39
790 VPOKE X1+40,130:VPOKE X1+880,130
800 NEXT X1
810 FOR Y1=2 TO 22
820 VPOKE Y1*40,130:VPOKE Y1*40+39,130
830 NEXT Y1
840 LOCATE 2,24,0
850 PRINT "JUGADOR DE LA IZQUIERDA :";F1;
860 LOCATE 22,24,0
870 PRINT "JUGADOR DE LA DERECHA :";F2;
880 X1=8:Y1=11:X2=32

890 Y2=11:H1=1:V1=0
900 H2=-1:V2=0:S1=0:S2=0
910 RETURN
920 FOR I=0 TO 23
930 READ A:VPOKE 3072+I,A*4
940 NEXT I
950 RETURN
960 DATA 30,43,53,43,53,43,53,30
970 DATA 30,63,63,63,63,63,63,30
980 DATA 63,63,63,63,63,63,63,63
```



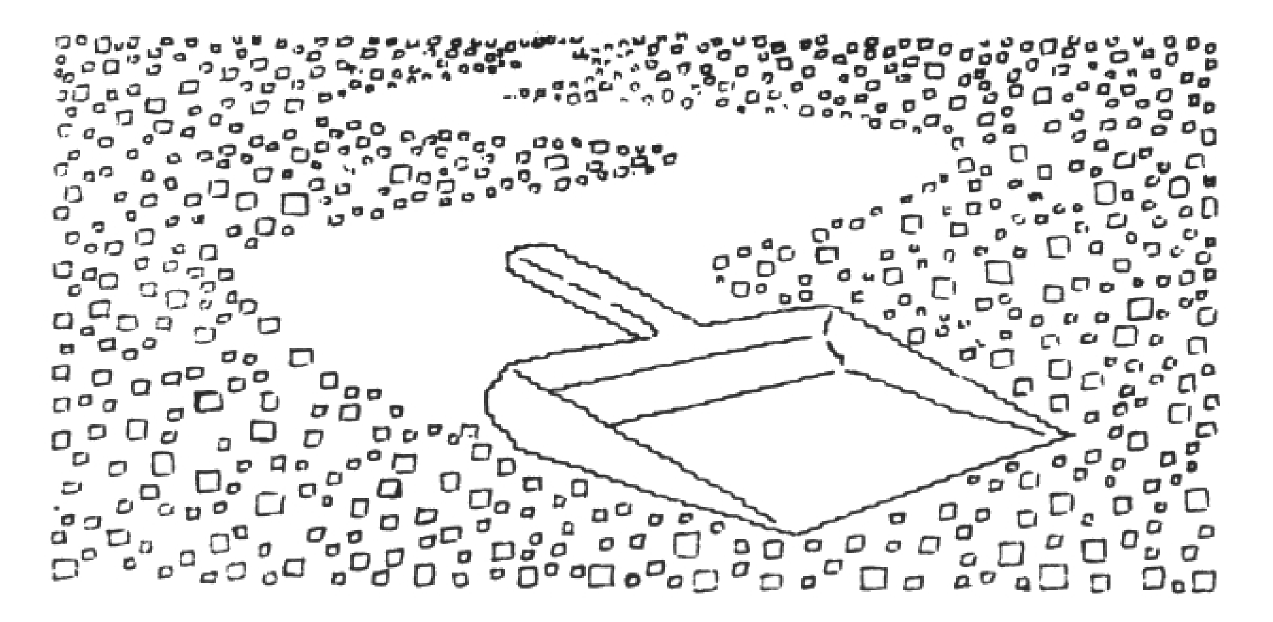


¡Dispóngase a practicar los deportes de invierno sin arriesgar a romperse una pierna! Láncese desde lo alto de la pista e intente pasar por el mayor número posible de puertas sin chocar con los palos. Para cambiar de dirección pulse cualquier tecla.

```
10 REM **********
20 REM * SLALOM *
30 REM **********
40 KEY OFF
50 WIDTH 39
60 GOSUB 450
70 FOR K=1 TO 300
80 X=SX+1
90 Y=SY
100 IF K>=16 AND (K-5)/10=INT((K-5)/10)
```

```
THEN GOSUB 370
110 IF K<284 AND K/10=INT(K/10) THEN GOS
UB 390
120 IF INKEY$<>"" THEN DX=-DX
130 S1=SX
140 SX=SX+DX
150 IF SX<2 THEN SX=2:DX=-DX
160 IF SX>36 THEN SX=36:DX=-DX
170 LOCATE S1,SY,0
180 PRINT B$;
190 LOCATE 0,24,0
200 PRINT
210 LOCATE SX,SY,0
220 PRINT S$(DX/2+.5);
230 S1=SX
240 NEXT K
250 LOCATE 5,5,0
260 PRINT "PUERTA(S) SALTADA(S):";T
270 LOCATE 5,8,0
280 PRINT "OTRA BAJADA ?"
290 D$=INKEY$
300 IF D$<>"" THEN 290
310 D$=INKEY$
320 IF D$="" THEN 310
330 IF D$<>"N" AND D$<>"n" THEN RUN
340 CLS
350 LOCATE 1,1,1
360 END
370 IF VPEEK(X-1+Y*40)=32 OR VPEEK(X+2+4
0*Y)=32 THEN IF VPEEK(X-2+Y*40)=32 OR VP
EEK(X+1+Y*40)=32 THEN T=T+1:BEEP
380 RETURN
390 P=P+(INT(RND(1)*3)-1)*8
400 IF P<2 THEN P=10
410 IF P>34 THEN P=26
420 LOCATE P,23
430 PRINT P$;
440 RETURN
450 DIM S$(1)
460 DEFINT A-Z
470 S$(0)=CHR$(128)
480 S$(1)=CHR$(129)
```

```
490 B$=CHR$(32)
500 P$=CHR$(130)+B$+B$+CHR$(131)
510 SCREEN 0,0
520 COLOR 4,15
530 DX=-1
540 SY=8
550 SX=19
560 S1=SX
570 P=18
580 GOSUB 660
590 CLS
600 LOCATE 5,10,0
610 PRINT "APRIETE CUALQUIER TECLA  PARA
EMPEZAR"
620 N=RND(1)
630 IF INKEY$="" THEN 620
640 CLS
650 RETURN
660 FOR I=0 TO 31
670 READ A
680 VPOKE 3072+I,A*4
690 NEXT I
700 RETURN
710 DATA 4,56,17,26,4,8,16,32
720 DATA 8,7,34,22,8,4,2,1
730 DATA 12,28,12,4,4,4,4,4
740 DATA 6,7,6,4,4,4,4,4
```



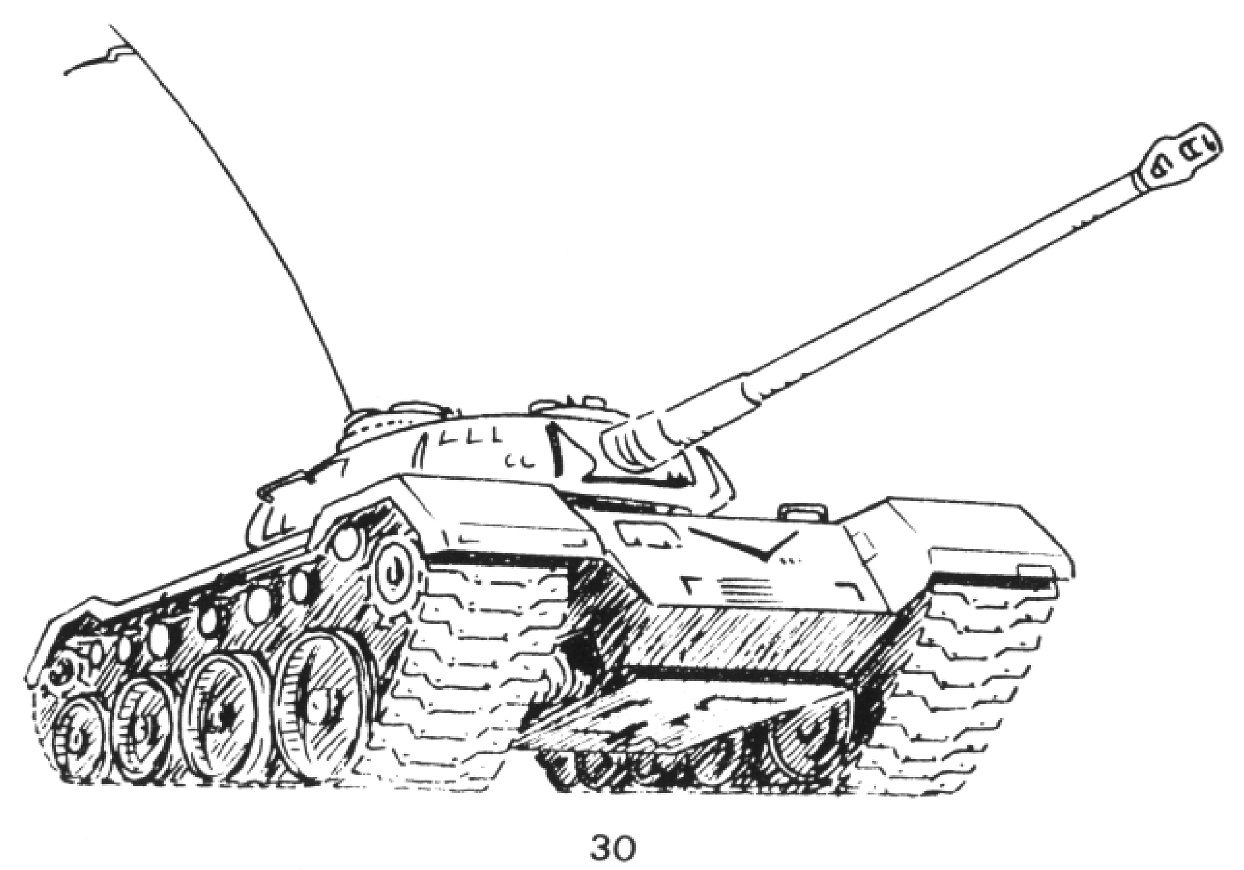


He aquí una forma curiosa de usar un ordenador; ha de intentar recoger lo más rápidamente posible las migajas que recubren el mantel. Dispone de 30 segundos para conseguir su limpieza total. Las migajas están representadas por puntos blancos. Las teclas de control del cursor, permitirán dirigir su recogedor.

```
10 REM *************
20 REM * RECOGEDOR *
30 REM *************
40 KEY OFF
50 GOSUB 260
60 LOCATE 0,23,0
70 PRINT "TIEMPO:";INT(Z);"PUNTUACION: ";S;
80 IF Z<1 THEN 580
90 K=STICK(0)
100 D1=(K=7)-(K=3)
110 D2=(K=1)-(K=5)
120 IF D1<>0 THEN DX=D1:DY=0
130 IF D2<>0 THEN DY=D2:DX=0
140 XP=PX+DX
150 YP=PY+DY
160 CR=VPEEK(XP+YP*40)
170 IF CR=J OR CR=B THEN XP=PX:YP=PY
180 IF CR=M THEN S=S+1:BEEP:X=X+1
190 VPOKE PX+PY*40,N
200 VPOKE XP+YP*40,J
```

```
210 PX=XP
220 PY=YP
230 Z=Z-.1
240 IF X=NM THEN 720
250 GOTO 60
260 SCREEN 0,0
270 COLOR 15,4,5
280 DEFINT A-Y
290 B=233
300 M=249
310 N=32
320 S=0
330 NM=10
340 X=0
350 FOR PX=0 TO 39
360 VPOKE PX+40,B
370 VPOKE 880+PX,B
380 NEXT PX
390 FOR PY=2 TO 21
400 VPOKE PY*40,B
410 VPOKE PY*40+39,B
420 NEXT PY
430 FOR I=1 TO NM
440 PX=INT(RND(-TIME)*37)+1
450 PY=INT(RND(-TIME)*21)+2
460 IF VPEEK(PX+PY*40)<>32 THEN 440
470 VPOKE PX+PY*40,M
480 NEXT I
490 PX=INT(RND(-TIME)*37)+1
500 PY=INT(RND(-TIME)*21)+2
510 IF VPEEK(PX+PY*40)<>32 THEN 490
520 VPOKE PX+PY*40,J
530 Z=30
540 DX=0
550 DY=0
560 J=219
570 RETURN
580 FOR I=1 TO 500
590 NEXT I
600 IF INKEY$<>"" THEN 600
610 IF S>R THEN R=S
```

```
620 LOCATE 13,10
630 PRINT "RECORD :";R;
640 LOCATE 13,16
650 PRINT " OTRA ?";
660 D$=INKEY$
670 IF D$="" THEN 660
680 IF D$<>"N" AND D$<>"n" THEN 50
690 CLS
700 LOCATE 0,0,1
710 END
720 NM=NM+1
730 VPOKE XP+YP*40,N
740 GOSUB 340
750 GOTO 60
```



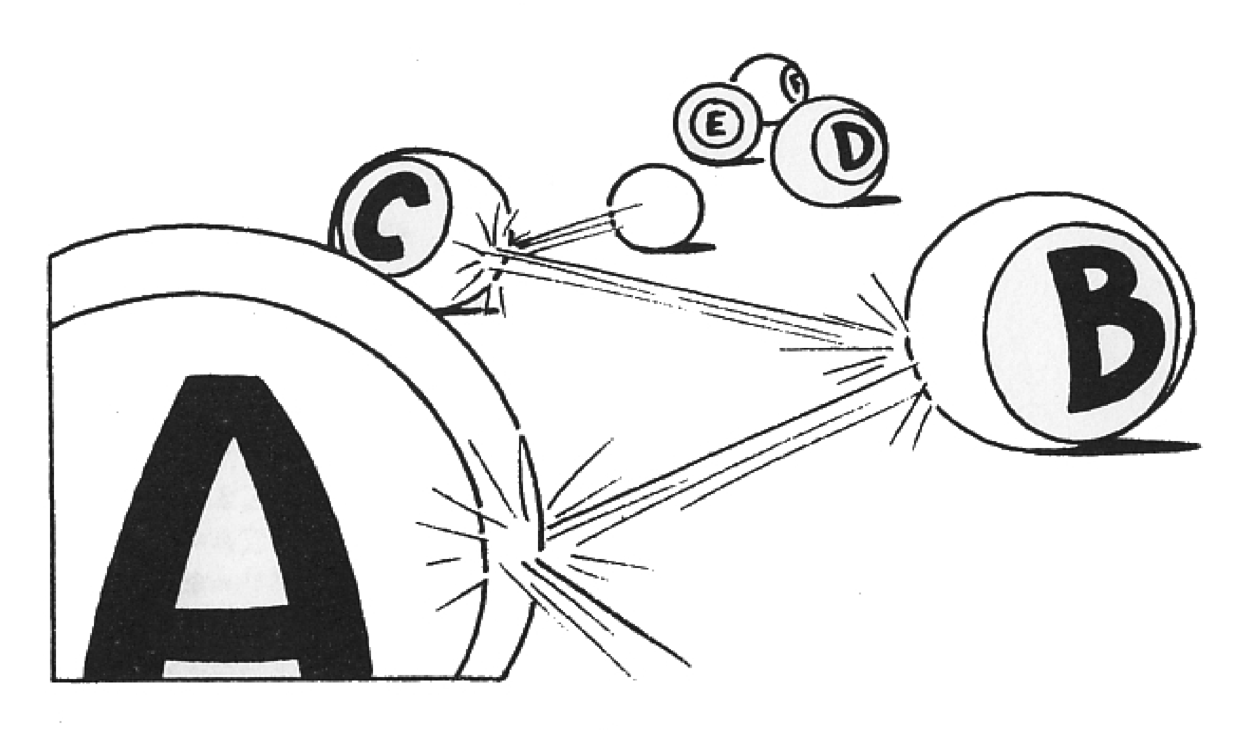


Procure avanzar con su tanque evitando las minas que el ordenador ha colocado al azar. Para dirigir el tanque hay que usar las teclas de control del cursor. Si el paso se halla bloqueado puede hacer explotar una mina disparando uno de sus diez proyectiles (pulsar la barra espaciadora).

```
10 REM ********
20 REM *TANQUE*
30 REM ********
40 KEY OFF
50 WIDTH 39
60 GOSUB 570
70 TX=TX+(STICK(0)=7)-(STICK(0)=3)
80 VPOKE X+40*Y,32
90 LOCATE 0,24,0
100 PRINT R$
110 IF VPEEK(TX+TY*40)<>32 THEN 300
120 VPOKE TX+TY*40,T
130 IF STRIG(0)=-1 THEN GOSUB 240
140 X=TX
150 Y=TY
160 PO=INT(RND(1)*7)+18
170 VPOKE 880+PO,M
180 PO=INT(RND(1)*7)+18
190 VPOKE 880+PO,M
200 S=S+1
210 FOR I=1 TO DL-S
220 NEXT I
230 GOTO 70
```

```
240 IF OB=0 THEN 290
250 OB=OB-1
260 FOR I=1 TO 5
270 VPOKE TX+(TY+I)*40,32
280 NEXT I
290 RETURN
300 FOR I=1 TO 10
310 VPOKE TX+TY*40,M
320 FOR J=1 TO 100
330 NEXT J
340 BEEP
350 VPOKE TX+TY*40,T
360 FOR J=1 TO 100
370 NEXT J
380 BEEP
390 NEXT I
400 IF S>R THEN R=S
410 CLS
420 LOCATE 10,10
430 PRINT "PUNTUACION ";S;
440 LOCATE 9,14
450 PRINT "RECORD :";R;
460 LOCATE 9,18,1
470 PRINT "OTRA ?";
480 IF INKEY$<>"" THEN 480
490 X$=INKEY$
500 IF X$="" THEN 490
510 IF X$<>"N" AND X$<>"n" THEN 550
520 CLS
530 COLOR 15,4
540 END
550 GOSUB 720
560 GOTO 70
570 CLS
580 SCREEN 0,0
590 COLOR 1,11
600 DEFINT A-Z
610 T=128
620 M=42
630 R$=""
640 FOR I=1 TO 15
650 R$=R$+CHR$(32)
```

```
660 NEXT I
670 R$=R$+CHR$(42)+CHR$(42)
680 FOR I=1 TO 7
690 R$=R$+CHR$(32)
700 NEXT I
710 R$=R$+CHR$(42)+CHR$(42)
720 CLS
730 TX=19
740 TY=10
750 X=TX
760 Y=TY
770 S=0
780 GOSUB 850
790 DL=200
800 OB=10
810 FOR I=1 TO 24
820 PRINT R$
830 NEXT I
840 RETURN
850 RESTORE
860 FOR I=0 TO 7
870 READ A
880 VPOKE 3072+I,A*4
890 NEXT I
900 RETURN
910 DATA 17,31,31,31,31,21,4,4
```


A
B
C
D
E



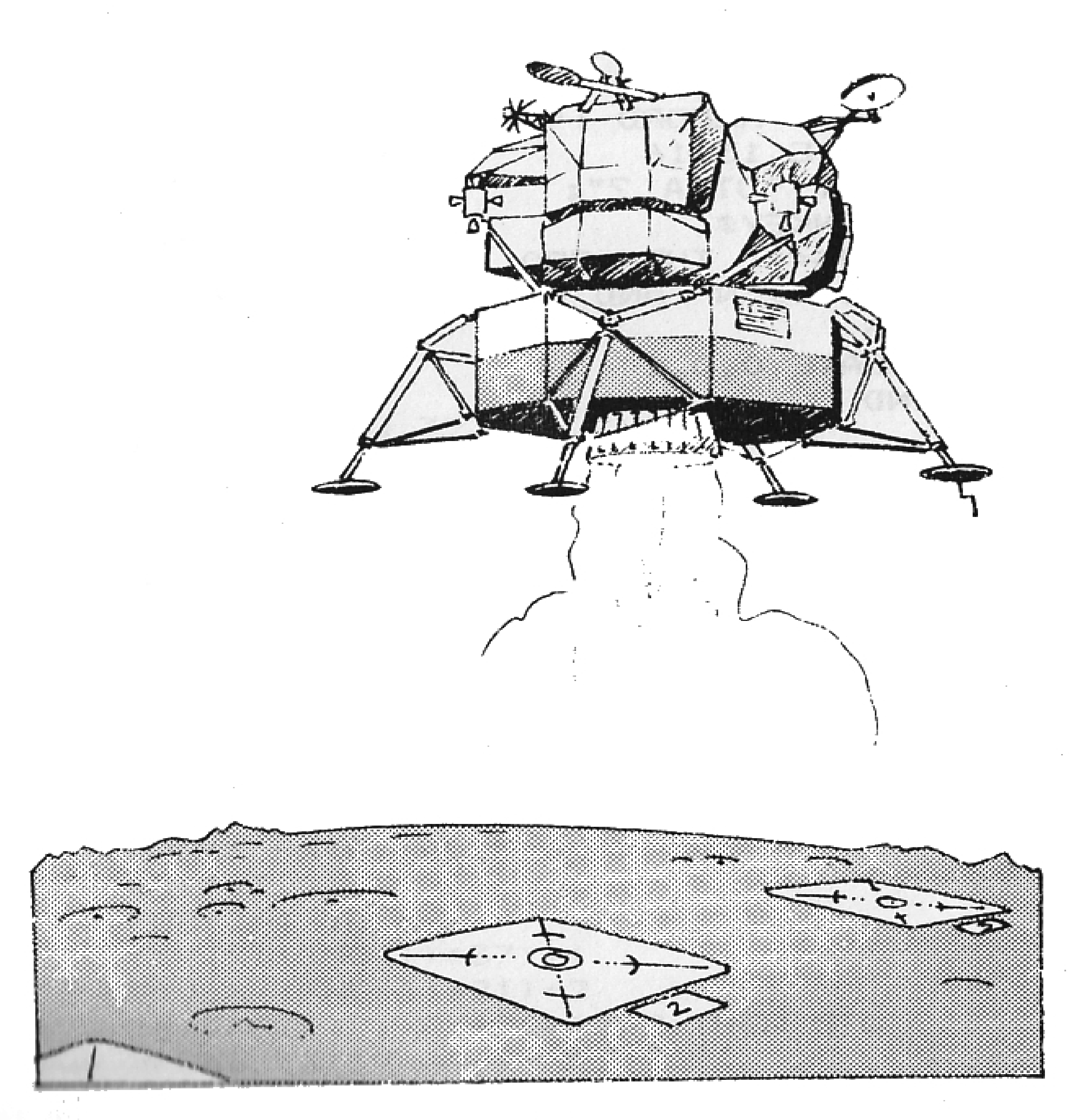
He aquí un juego que requiere bastante destreza. Debe intentar marcar el mayor número posible de puntos, eliminando las letras que son presentadas por la computadora. Utilice las teclas de control del cursor.

Atención: debe eliminar las letras por orden alfabético, evitando los obstáculos que se encuentran dispuestos al azar sobre la pantalla, y ello dentro del tiempo límite. (El tiempo que queda aparece en la parte baja de la pantalla). Cuando todas las letras estén eliminadas, el juego continúa con una letra suplementaria.

```
10 REM ************
20 REM * ALFABETO *
30 REM ************
40 KEY OFF
50 WIDTH 39
60 GOSUB 350
70 FOR I=1 TO X
80 LOCATE 0,23,0
90 PRINT "TEMPS :";INT(Z),"SCORE :";S;
100 IF Z<1 THEN 700
110 K=STICK(0)
120 D1=(K=7)-(K=3)
130 D2=(K=1)-(K=5)
140 IF D1<>0 THEN DX=D1:DY=0
150 IF D2<>0 THEN DY=D2:DX=0
160 XP=PX+DX
170 YP=PY+DY
180 CR=VPEEK(XP+YP*40)
```

```
190 IF CR=I+64 THEN S=S+1:BEEP:GOTO 270
200 IF CR<>32 THEN XP=PX:YP=PY
210 VPOKE PX+PY*40,N
220 VPOKE XP+YP*40,J
230 PX=XP
240 PY=YP
250 Z=Z-.1
260 GOTO 80
270 VPOKE PX+PY*40,N
280 VPOKE XP+YP*40,J
290 PX=XP
300 PY=YP
310 Z=Z-.1
320 NEXT I
330 GOSUB 350
340 GOTO 70
350 SCREEN 0,0
360 COLOR 15,3
370 DEFINT A-Y
380 J=215
390 M=219
400 X=X+1
410 N=32
420 FOR PX=0 TO 39
430 VPOKE PX+40,M
440 VPOKE 880+PX,M
450 NEXT PX
460 FOR PY=2 TO 21
470 VPOKE PY*40,M
480 VPOKE PY*40+39,M
490 NEXT PY
500 FOR I=1 TO 120
510 PX=INT(RND(1)*37)+1
520 PY=INT(RND(1)*21)+2
530 IF VPEEK(PX+PY*40)<>32 THEN 510
540 VPOKE PX+PY*40,M
550 NEXT I
560 FOR Z=1 TO X
570 PX=INT(RND(1)*37)+1
580 PY=INT(RND(1)*21)+2
590 IF VPEEK(PX+PY*40)<>32 THEN 570
600 VPOKE PX+PY*40,Z+64
```

```
610 NEXT Z
620 PX=INT(RND(-TIME)*37)+1
630 PY=INT(RND(-TIME)*21)+2
640 IF VPEEK(PX+PY*40)<>32 THEN 620
650 VPOKE PX+PY*40,J
660 Z=50
670 DX=0
680 DY=0
690 RETURN
700 FOR I=1 TO 500
710 NEXT I
720 IF INKEY$<>"" THEN 720
730 IF S>R THEN R=S
740 LOCATE 13,10
750 PRINT "RECORD :";R;
760 LOCATE 13,16
770 PRINT "¿OTRA ?";
780 D$=INKEY$
790 IF D$="" THEN 780
800 IF D$<>"N" AND D$<>"n" THEN RUN
810 CLS
820 LOCATE 0,0,1
830 END
```


2

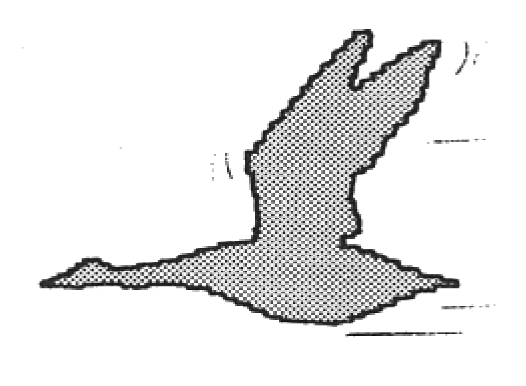


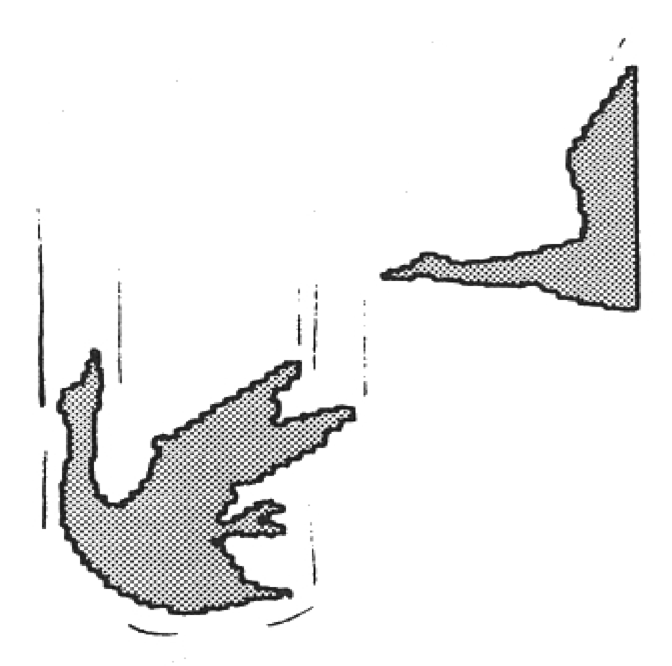
Después de un largo viaje sin gravedad, no resulta nada fácil aterrizar suavemente con una nave espacial; pero gracias a su ordenador, está en condiciones para efectuar un entrenamiento sin riesgos. Debe colocar su nave sobre la zona destinada al efecto. Puede dirigirse hacia la derecha y hacia la izquierda con ayuda de las teclas <→> y <←>, tras cada aterrizaje con éxito, el juego vuelve a comenzar con mayor velocidad. La cantidad de combustible disponible viene indicada por la longitud del trazo horizontal que aparece en la parte alta de la pantalla.

```
10 REM ****************
20 REM *  ATERRIZAJE   *
30 REM ****************
40 GOSUB 350
50 GOSUB 520
60 FOR NY=1 TO 21
70 D$=INKEY$
80 NH=NX
90 NX=NX+(STICK(0)=7)-(STICK(0)=3)
100 IF NX<2 THEN NX=2
110 IF NX>36 THEN NX=36
120 LOCATE NH,NY-1:PRINT B2$;
```

```
130 LOCATE NH,NY:PRINT B2$;
140 LOCATE NX,NY:PRINT N1$;
150 LOCATE NX,NY+1:PRINT N2$;
160 FOR I=1 TO DL:NEXT I
170 NEXT NY
180 IF INT((NX-4)/10)=(NX-4)/10 THEN GOT
O 50
190 LOCATE NX,NY:PRINT B2$;
200 LOCATE NX,NY-1:PRINT B2$;
210 IF NX<5 THEN NX=5
220 LOCATE NX-3,NY:PRINT NC$;
230 BEEP
240 FOR I=1 TO 400:NEXT I
250 LOCATE 8,10:PRINT "LA NAVE SE HA
ESTRELLADO";
260 LOCATE 15,12:PRINT "PUNTUACION :";S-1;
270 LOCATE 14,15:PRINT "OTRA ?";
280 IF INKEY$<>"" THEN 280
290 D$=INKEY$
300 IF D$="" THEN 290
310 IF D$<>"N" AND D$<>"n" THEN RUN
320 LOCATE 0,0,1
330 CLS
340 END
350 COLOR 10,4
360 SCREEN 0,0
370 WIDTH 39
380 KEY OFF
390 FOR I=0 TO 39
400 READ A
410 VPOKE 3072+I,A*4
420 NEXT I
430 B$=CHR$(32)
440 B2$=B$+B$
450 N1$=CHR$(128)+CHR$(129)
460 N2$=CHR$(130)+CHR$(131)
470 A$=CHR$(132)+CHR$(132)
480 NC$=CHR$(131)+CHR$(129)+B$+CHR$(128)
+CHR$(130)
490 S=0
500 DL=100
510 RETURN
```

```
520 FOR I=1 TO 500:NEXT I
530 CLS
540 FOR I=0 TO 3
550 LOCATE 4+I*10,24,0:PRINT A$;
560 NEXT I
570 DL=DL-10
580 IF DL<1 THEN DL=1
590 S=S+1
600 NX=INT(RND(1)*35)+2
610 RETURN
620 DATA 0,7,15,31,59,59,63,59
630 DATA 0,56,60,62,55,55,63,55
640 DATA 59,31,15,18,32,32,16,56
650 DATA 55,62,60,18,1,1,2,7
660 DATA 63,63,63,0,0,0,0,0
```







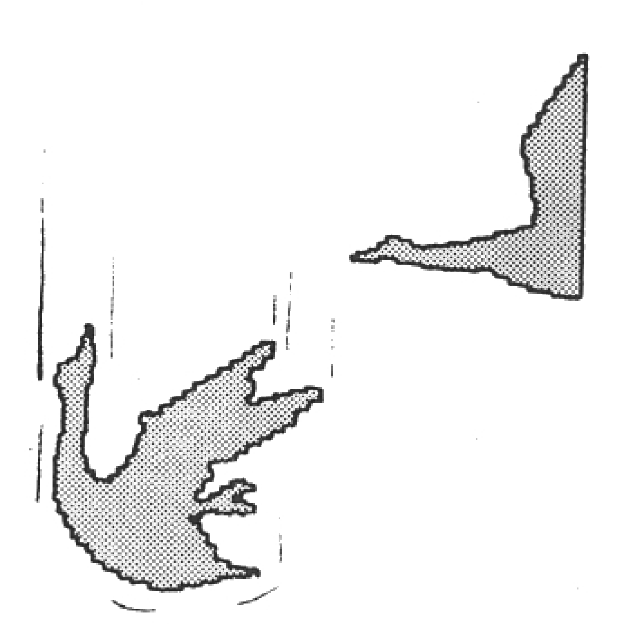
Su ordenador le invita a una cacería de patos. Los patos vuelan de derecha a izquierda por la parte alta de la pantalla. Vd. se desplaza utilizando las teclas del cursor. Puede efectuar tantos disparos como quiera, pero tan sólo pasarán 20 patos y su objetivo es abatir el mayor número posible. Para disparar, pulse la barra espaciadora. Si alcanza un pato, conseguirá un punto y lo verá caer batiendo sus alas.

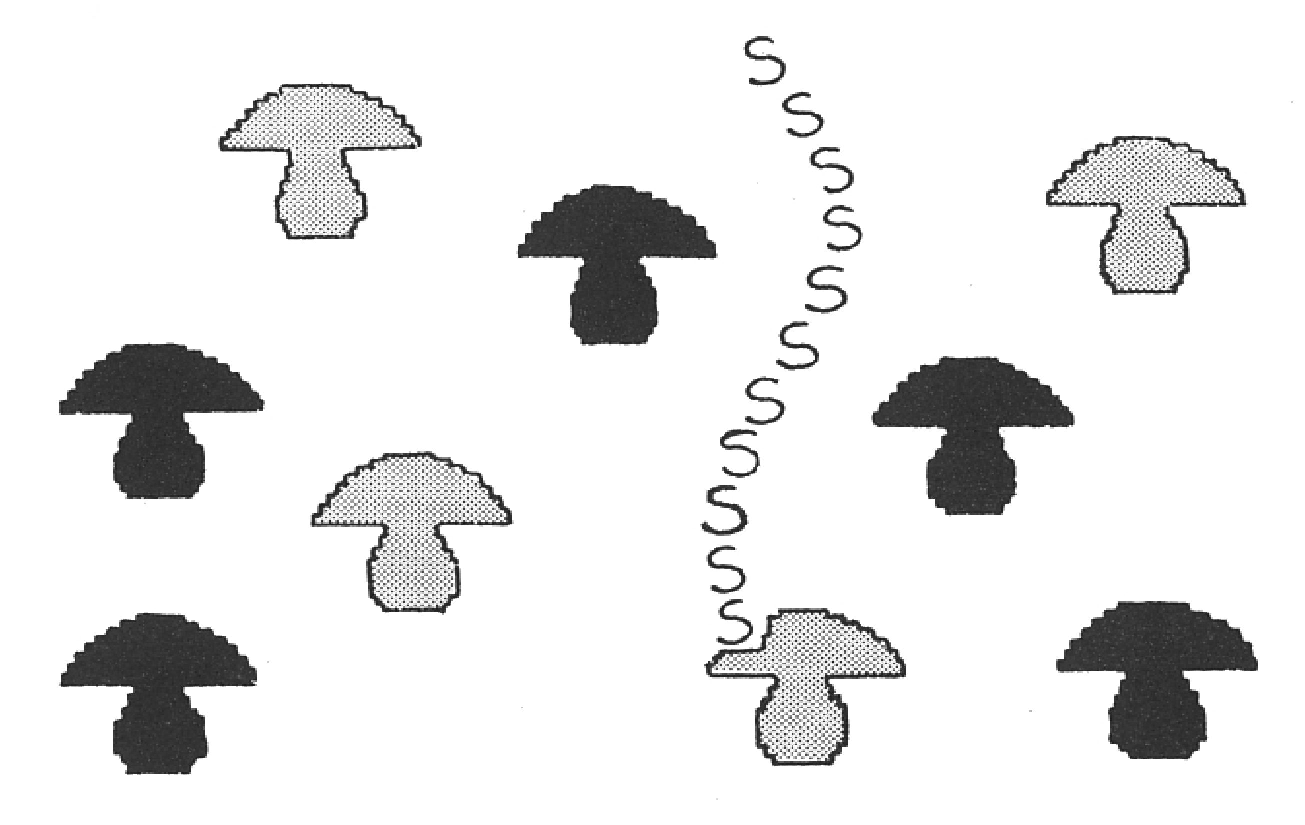
```
10 REM **********************
20 REM * LA CAZA DEL PATO *
30 REM **********************
40 KEY OFF
50 WIDTH 39
60 GOSUB 330
70 Z=-Z
80 H=Z+.5
90 CX=CX-1
100 IF CX<0 THEN GOSUB 560
110 IF NC=0 THEN 710
120 LOCATE CX,CY,0
```

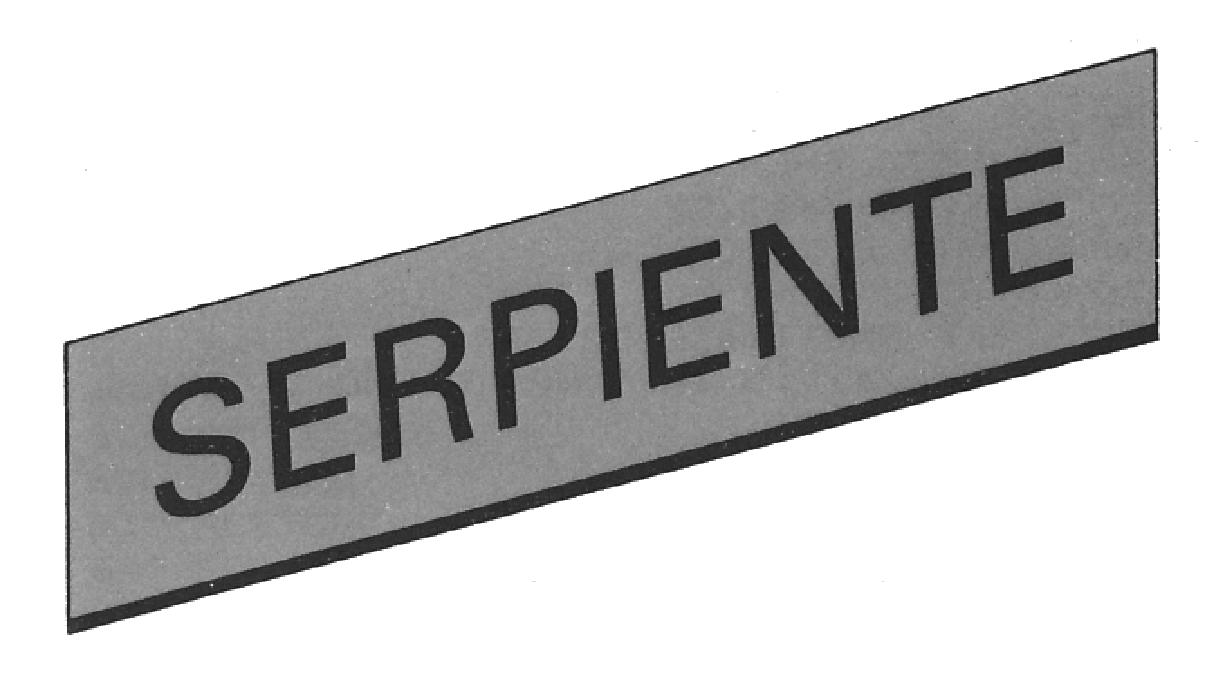
```
130 PRINT CC$(H);
140 LOCATE X,Y,0
150 PRINT N$;
160 LOCATE JX,JY
170 PRINT J$;
180 X=JX
190 IF TY<>JY THEN TY=TY-2:GOTO 210
200 TX=JX
210 IF TY<2 THEN 610
220 LOCATE TX,TY,0
230 PRINT J$;
240 IF TY<>JY THEN LOCATE TX,TY+2,0:PRIN
T N$;
250 JX=JX+2*((STICK(0)=7)-(STICK(0)=3))
260 IF JX>37 THEN JX=37
270 IF JX<2 THEN JX=2
280 IF STRIG(0)=0 THEN 70
290 IF TY<>JY THEN 70
300 TY=JY-2
310 TX=JX
320 GOTO 70
330 CLS
340 DEFINT A-Y
350 SCREEN 0,0
360 GOSUB 1110
370 COLOR 3,4
380 CLS
390 DIM CC$(1)
400 J$=CHR$(128)
410 N$=CHR$(32)
420 E$=CHR$(42)
430 CC$(0)=CHR$(129)+CHR$(130)+CHR$(131)
+N$
440 CC$(1)=CHR$(129)+CHR$(132)+CHR$(131)
+N$
450 JX=19
460 JY=24
470 NC=20
480 TX=JX
490 TY=JY
500 Z=.5
510 CX=36
```

```
520 CY=0
530 X=JX
540 Y=JY
550 RETURN
560 LOCATE 0,0,0
570 PRINT N$+N$+N$;
580 CX=36
590 NC=NC-1
600 RETURN
610 IF TX=CX OR TX=CX+1 THEN 830
620 LOCATE TX,TY+2,0
630 PRINT N$;
640 LOCATE TX,TY,0
650 PRINT J$;
660 LOCATE TX,TY,0
670 PRINT N$;
680 TX=JX
690 TY=JY
700 IF NC<>0 THEN 220
710 CLS
720 LOCATE 10,10,0
730 PRINT "PUNTUACION:";S;
740 LOCATE 9,20,0
750 PRINT "OTRA ?";
760 IF INKEY$<>"" THEN 760
770 X$=INKEY$
780 IF X$="" THEN 770
790 IF X$<>"N" AND X$<>"n" THEN RUN
800 LOCATE 0,0,1
810 CLS
820 END
830 LOCATE TX,TY,0
840 PRINT E$;
850 LOCATE TX,TY+2,0
860 PRINT N$;
870 BEEP
880 FOR I=1 TO 50
890 NEXT I
900 S=S+1
910 DX=CX
920 CX=36
930 FOR I=1 TO 24
```

```
940 Z=-Z
950 H=Z+.5
960 LOCATE DX,I-1,0
970 PRINT N$+N$+N$;
980 LOCATE DX,I,0
990 PRINT CC$(H);
1000 BEEP
1010 FOR J=1 TO 10
1020 NEXT J
1030 NEXT I
1040 NC=NC-1
1050 FOR I=1 TO 500
1060 NEXT I
1070 CLS
1080 IF NC=0 THEN 710
1090 TY=JY
1100 GOTO 70
1110 FOR I=0 TO 39
1120 READ A
1130 VPOKE 3072+I,A*4
1140 NEXT I
1150 RETURN
1160 DATA 4,14,31,4,4,4,4,4
1170 DATA 0,0,0,25,63,0,0,0
1180 DATA 31,60,56,63,63,0,0,0
1190 DATA 0,0,0,56,32,0,0,0
1200 DATA 0,0,0,63,63,56,60,31
```





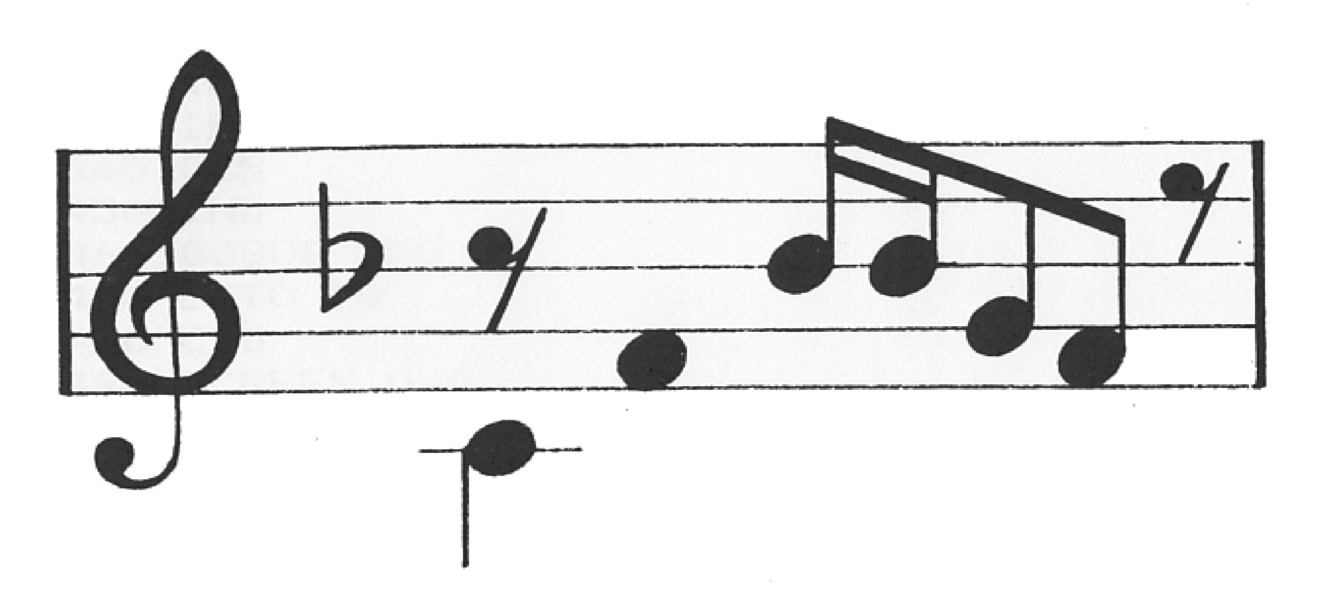


En este juego, Vd. es una serpiente que se desplaza contorneándose sobre la pantalla. El cambio de dirección se consigue pulsando cualquier tecla. Para poder desplazarse es preciso que se alimente. Felizmente se halla rodeado por cantidad de setas. Pero, ¡cuidado! Si bien algunos son excelentes, ha de evitar las setas vacías, puesto que son venenosas. Cada seta azul le proporciona las calorías necesarias para avanzar diez líneas. ¡Procure no morirse de hambre pero sin acabar envenenado!

```
10 REM ***********
20 REM *SERPIENTE*
30 REM ***********
40 KEY OFF
50 WIDTH 39
60 GOSUB 480
70 D$=INKEY$
80 IF D$<>"" THEN D=-D
90 X=X+D
100 IF X<1 THEN X=2
110 IF X>39 THEN X=38
120 C=VPEEK(X+40*Y)
```

```
130 IF C=PQ THEN 240
140 IF C=TR THEN S=S+10:H=H+10:BEEP
150 VPOKE X+Y*40,SP
160 LOCATE 0,24,0
170 PRINT
180 VPOKE RND(1)*38+921,TR
190 IF RND(1)>.5 THEN VPOKE RND(1)*38+92
1,PQ
200 S=S-1
210 IF S=0 THEN 240
220 H=H+1
230 GOTO 70
240 BEEP
250 LOCATE 0,24,0
260 PRINT
270 VPOKE X+(Y-1)*40,SP
280 FOR I=1 TO 5
290 BEEP
300 FOR J=1 TO 50
310 NEXT J
320 NEXT I
330 IF INKEY$<>"" THEN 330
340 IF H>R THEN R=H
350 LOCATE 4,20
360 PRINT "PUNTUACION :";H;
370 LOCATE 23,20
380 PRINT "RECORD :";R;
390 LOCATE 15,23,1
400 PRINT "OTRA ?";
410 D$=INKEY$
420 IF D$="" THEN 410
430 IF D$<>"N" AND D$<>"n" THEN 460
440 CLS
450 END
460 GOSUB 550
470 GOTO 70
480 CLS
490 SCREEN 0,0
500 DEFINT A-Z
510 COLOR 5,3
520 SP=128
530 PQ=130
```

```
540 TR=129
550 CLS
560 S=1000
570 H=0
580 D=1
590 X=19
600 Y=10
610 GOSUB 630
620 RETURN
630 RESTORE
640 FOR I=0 TO 23
650 READ A
660 VPOKE 3072+I,A*4
670 NEXT I
680 RETURN
690 DATA 51,63,45,63,63,30,12,0
700 DATA 30,63,63,30,12,12,30,0
710 DATA 30,33,33,30,12,12,30,0
```





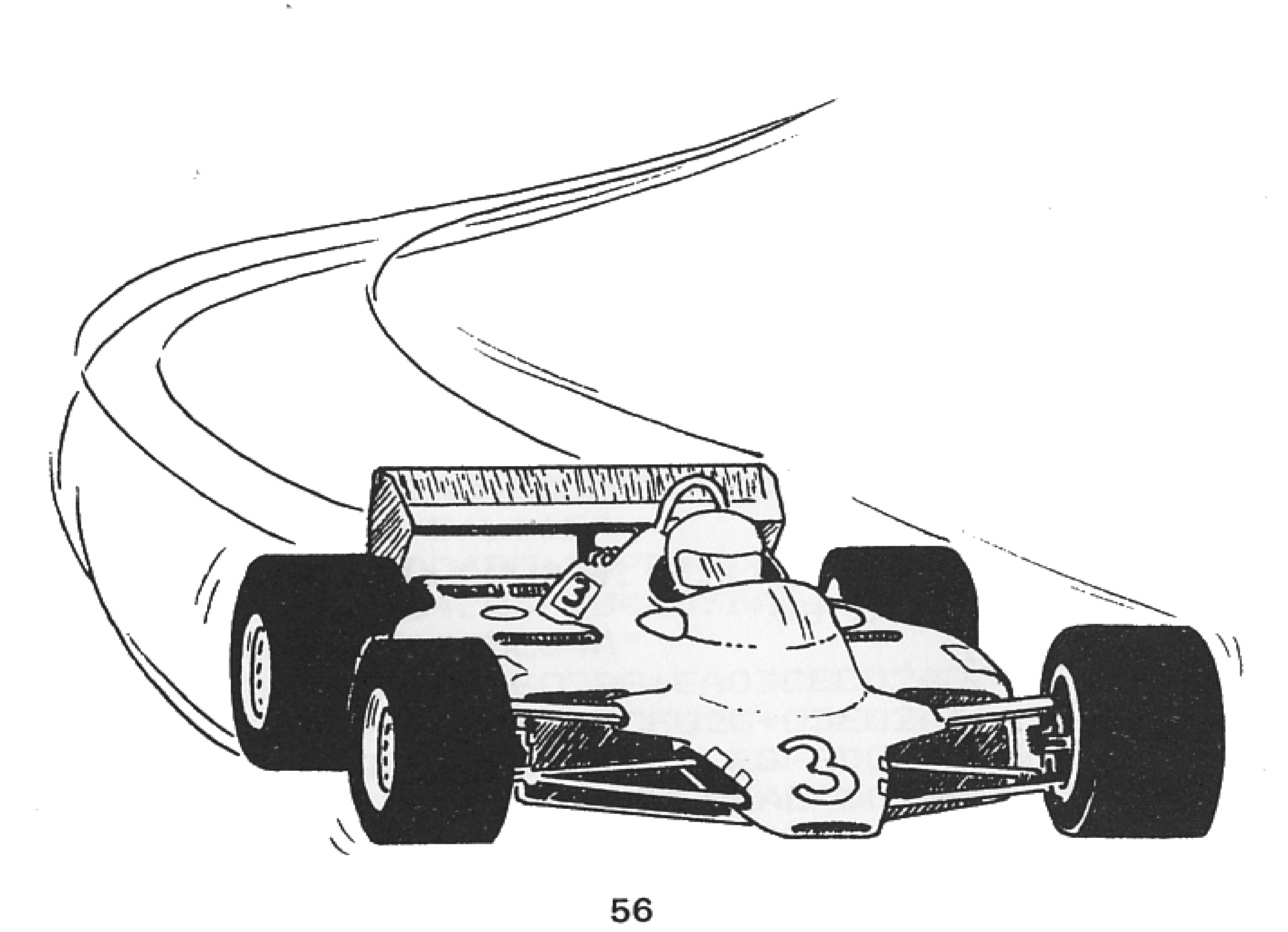
Este programa no es realmente un juego pero sí es una excelente demostración de las amplias posibilidades musicales que tiene el M.S.X.

Para esta composición en "La menor" de Juan Sebastian Bach; sólo se emplean dos voces. La labor es poner la tercera.

```
10 REM ***********
20 REM *  MUSICA  *
30 REM ***********
40 KEY OFF
50 CLEAR 5000
60 SCREEN 1,2
70 A$="V5T32L32O4R32EAO5CO4BEBO5DL16CEO4
G#O5E"
80 A1$="V5T32L32O2A16O3A8G+16AEAO4CO3BEB
O4D"
90 A$=A$+"O4L32AEAO5CO4BEBO5DL16CO4A8R16
"
100 A1$=A1$+"L16CO3AG+EL32AEAO4CO3BEBO4D
"
110 A$=A$+"O5L32R32ECEO4AO5CO4EGL16FAO5D
FL32FD"
120 A1$=A1$+"L16CO3AO4CO3AL32O4DO3AFADFO
2AO3CO2L16B"
130 B$="O4BO5DO4GBDFL16EGO5CE"
140 B1$="O3DGBL32BGEGCEO2GBL16AO3CL32DFO
2BO3DO2L16GBO3L32CEO2AO3CO2L16FDL32GO3GF
GCGO4CEDO3GO4DF"
150 B$=B$+"L32ECO4AO5CO4L16FO5DL32DO4BGB
L16EO5CL32CO4AFAL16DBO5C8R8"
```

```
160 B$=B$+"O4L32R32GO5CEDO4GO5DFL16EGO4B
O5GL32CO4GO5CEDO4G"
170 B1$=B1$+"L16ECO3BGL32O4CO3GO4CEDO3GO
4DFL16EC8"
180 C$="O5DFL16ECGE"
190 C1$="R16R32L32GEGCEO3GB"
200 C$=C$+"O6L32CO5AEACEO4AO5CL16DF+AO6C
"
210 C1$=C1$+"L16AO4CEGL32F+ADF+O3AO4DO3F
+A"
220 C$=C$+"L32O5BGDGO4BO5DO4GBO5L16CEGBL
32AF+"
230 C1$=C1$+"L16GBO4DF+L32EGCEO3GO4CO3EG
L16F+"
240 D$="D+F+O4BO5D+O4F+AL16GO5GL32GECE"
250 D1$="ABO4D+L32R32ECEO3AO4CEG"
260 D$=D$+"O4L16AO5F+L32F+DO4BO5DO4L16GO
5EL32ECO4AO5C"
270 D1$=D1$+"F+DO3BO4DO3GBO4DF+ECO3AO4CO
3F+AO4C16"
280 D$=D$+"O4F+O5GF+ED+F+O4BO5D+E8R8L32"
290 D1$=D1$+"CO3BO4CO3AL16BO2BO3L32EO4EO
3BGEO2BGB"
300 E$="R32GB-GEGC+EGEC+EO4AGFEDO5FA-FDF
O4BO5DFDO4BO5DO4GFEDCO5EGECEO4AO5CD+CO4A
O5CO4F+ED+C+O3BO5D"
310 E1$="L16EO3EGB-C+8R8O4DO3DFA-O2B8R8O
4CO3CEF+O2A8R8O3B"
320 F$="FDO4BO5DO4G+BO5DO4BG+BER32R16R32
EAO5CO4BEBO5DL16CO4AG+EL32AO5CECO4AO5CO4
F+AO5CO4AF+AD+O5CO4BAG+B"
330 F1$="O2BO3DFO2G+8R32O4L32DCO3BO4L16C
O3AG+EL32AEAO4CO3BEBO4DCEAECEO3AO4CO3F+A
O4CO3AF+AD+F+L16E"
340 G$="O5DO4BG+BDFG+FDFO3BO4FEDCEAECEO3
AO4CD+CO3AO4CO3F+O4CO3BAL16G+O4BG+ER32L3
2EAO5CO4BEBO5DCO4A"
350 G1$="G+BG+EO2BG+EAO3CECO2AO3CO2D+8R3
2L32O3BG+EDBG+DL16CEO2G+O3EO2A"
360 H$="O5CEDO4BO5DFECEGFEDCO4BO5CDEFDG+
DBDCAFDO4BO5DO4G+BO5CO4AEABG+AECEO3A8"
370 H1$="O3F+O2BO3G+CADB-G+FDO2BG+ADEFD+
```

```
E03E02A8A8"
380 FOR I=1 TO 10
390 X$=X$+CHR$(0)
400 NEXT I
410 X$=X$+CHR$(30)+CHR$(63)+CHR$(127)+CH
R$(127)+CHR$(63)+CHR$(30)
420 X$=X$+CHR$(192)+CHR$(160)+CHR$(144)+
CHR$(144)+CHR$(144)+CHR$(160)
430 FOR I=1 TO 8
440 X$=X$+CHR$(128)
450 NEXT I
460 X$=X$+CHR$(0)+CHR$(0)
470 SPRITE$(1)=X$
480 GOSUB 660
490 PLAY A$,A1$
500 GOSUB 660
510 PLAY B$,B1$
520 GOSUB 660
530 PLAY C$,C1$
540 GOSUB 660
550 PLAY D$,D1$
560 GOSUB 660
570 PLAY E$,E1$
580 GOSUB 660
590 PLAY F$,F1$
600 GOSUB 660
610 PLAY G$,G1$
620 GOSUB 660
630 PLAY H$,H1$
640 GOSUB 660
650 GOTO 490
660 FOR I=1 TO 15
670 X=RND(1)*240
680 Y=RND(1)*176
690 PUT SPRITE I,(X,Y),RND(1)*15,1
700 FOR J=1 TO 80
710 NEXT J
720 NEXT I
730 RETURN
```



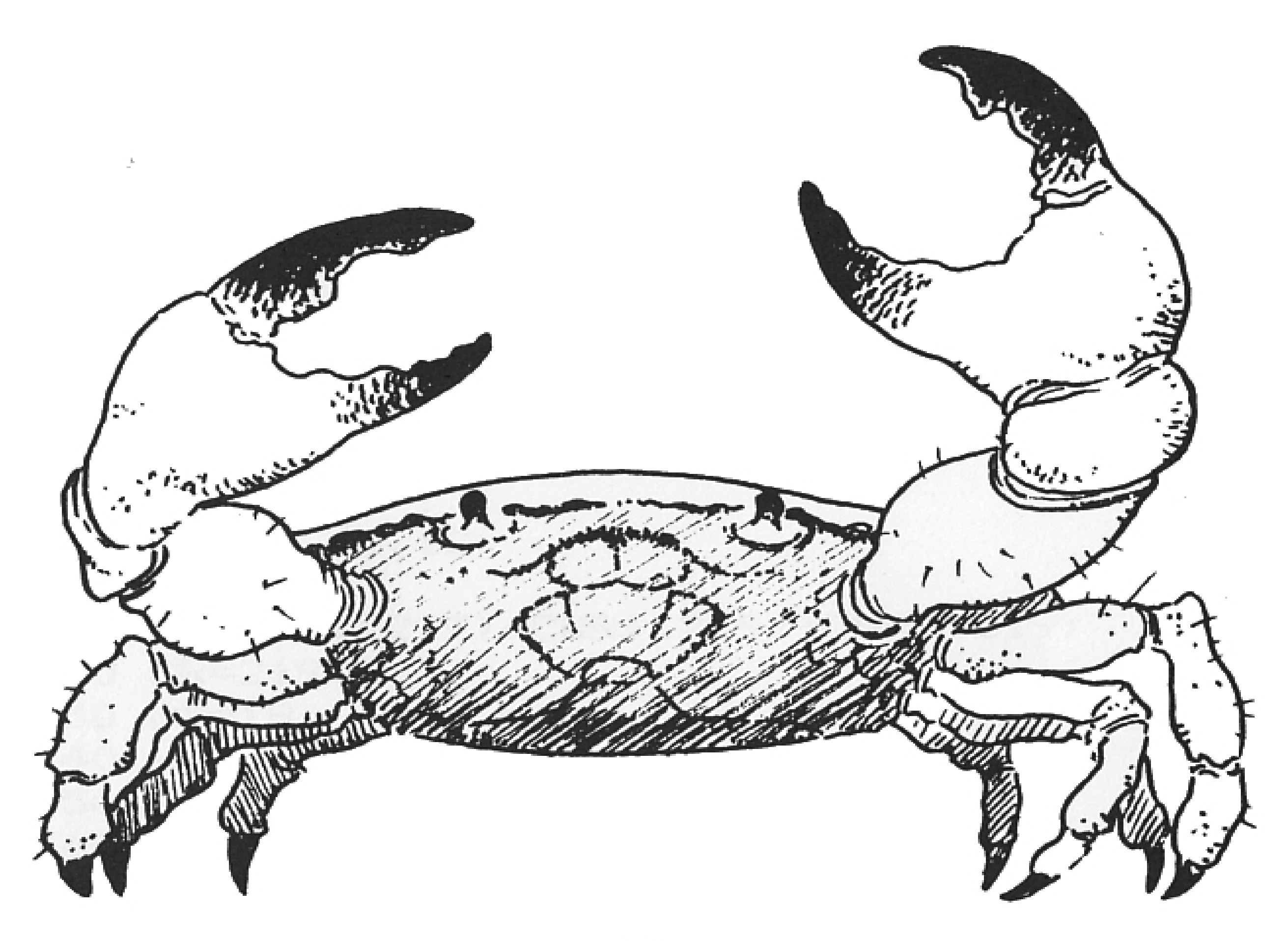


Al volante de su fórmula 1, intente recorrer la mayor distancia posible. Su coche dispone de dos velocidades. Con la barra espaciadora apretada se mantendrá en segunda velocidad. El volante se conduce con las teclas de dirección del cursor. En su segunda velocidad, el vehículo marcha a doble velocidad. Pero, ¡cuidado con el accidente!

```
10 REM **************
20 REM * GRAN PREMIO *
30 REM **************
40 KEY OFF
50 WIDTH 39
60 GOSUB 450
70 H=STICK(0)
80 PX=PX+(H=7)-(H=3)
90 IF STRIG(0) THEN T=2 ELSE T=1
100 IF VPEEK(PX+40)<>CR THEN 240
110 RX=RX+(RND(1)<.5)-(RND(1)<.5)
120 IF RX<RN THEN RX=RN
130 IF RX>RM THEN RX=RM
140 VPOKE XP,CR
150 LOCATE RX,24,0
```

```
160 PRINT R$
170 VPOKE PX,V
180 K=K+T
190 DL=(2-T)*50
200 FOR I=1 TO DL
210 NEXT I
220 XP=PX
230 GOTO 70
240 VPOKE XP,CR
250 FOR I=1 TO 5
260 VPOKE PX+40,A
270 FOR J=1 TO 100
280 NEXT J
290 BEEP
300 VPOKE PX+40,V
310 FOR J=1 TO 100
320 NEXT J
330 NEXT I
340 LOCATE 10,10,0
350 PRINT "KMS RECORRIDOS:";K;
360 IF INKEY$<>"" THEN 360
370 LOCATE 14,16,0
380 PRINT "OTRA ?";
390 D$=INKEY$
400 IF D$="" THEN 390
410 IF D$<>"N" AND D$<>"n" THEN RUN
420 LOCATE 0,0,1
430 CLS
440 END
450 CLS
460 SCREEN 0,0
470 GOSUB 680
480 DEFINT A-Y
490 COLOR 14,12
500 A=221
510 R$=CHR$(219)+CHR$(219)+CHR$(219)
520 CR=219
530 RX=18
540 V=128
550 RY=24
560 T=1
570 PX=420
```

```
580 XP=PX
590 LOCATE 0,0,0
600 FOR I=0 TO 24
610 LOCATE RX,I,0
620 PRINT R$
630 NEXT I
640 VPOKE PX,V
650 RM=34
660 RN=2
670 RETURN
680 FOR I=0 TO 7
690 READ A
700 VPOKE 3072+I,A*4
710 NEXT I
720 RETURN
730 DATA 18,0,18,51,51,18,0,18
```





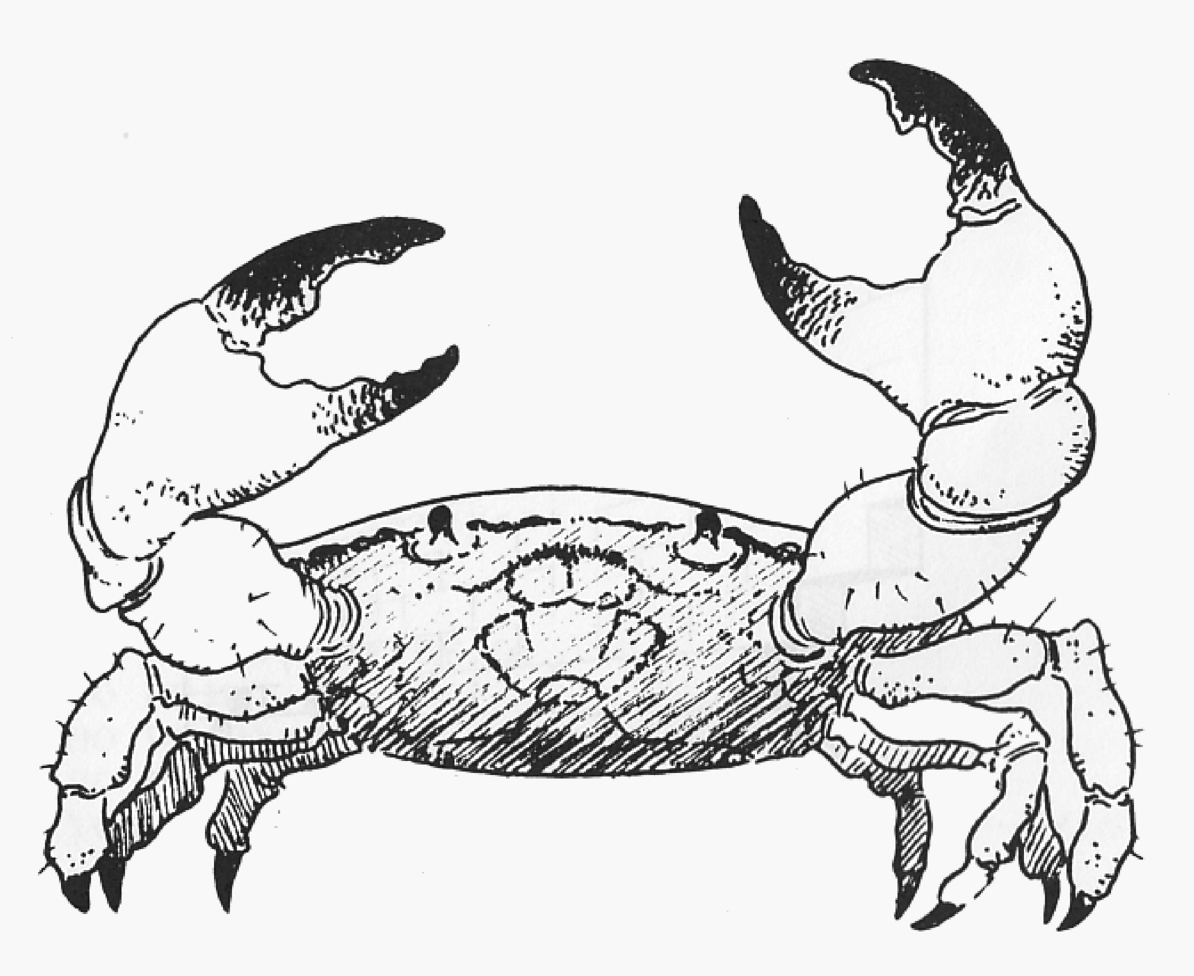
Ahora debe ayudar a una pobre tortuga a volver al mar, evitando a los voraces cangrejos que patrullan por la playa. Cada tortuga que alcance su propósito le proporciona 1 punto. Dispone de cinco itinerarios para intentar marcar su puntuación máxima. Emplee las teclas de control del cursor para avanzar y para retroceder.

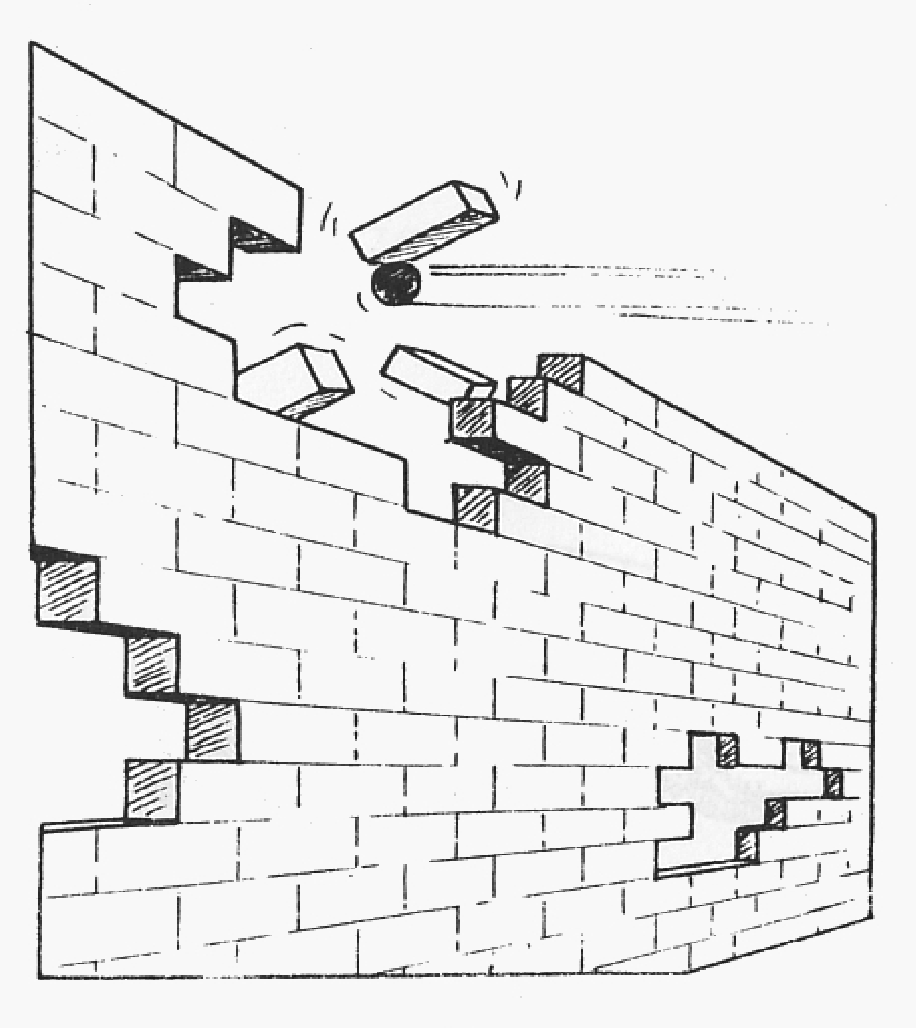
```
10 REM *************
20 REM * CANGREJOS *
30 REM *************
40 SCREEN 0,0
50 DEFINT A-Z
60 KEY OFF
70 WIDTH 39
80 GOSUB 990
90 GOSUB 1130
100 GOSUB 840
110 LOCATE 0,20,0
120 PRINT "VIDAS RESTANTES.";NP;
```

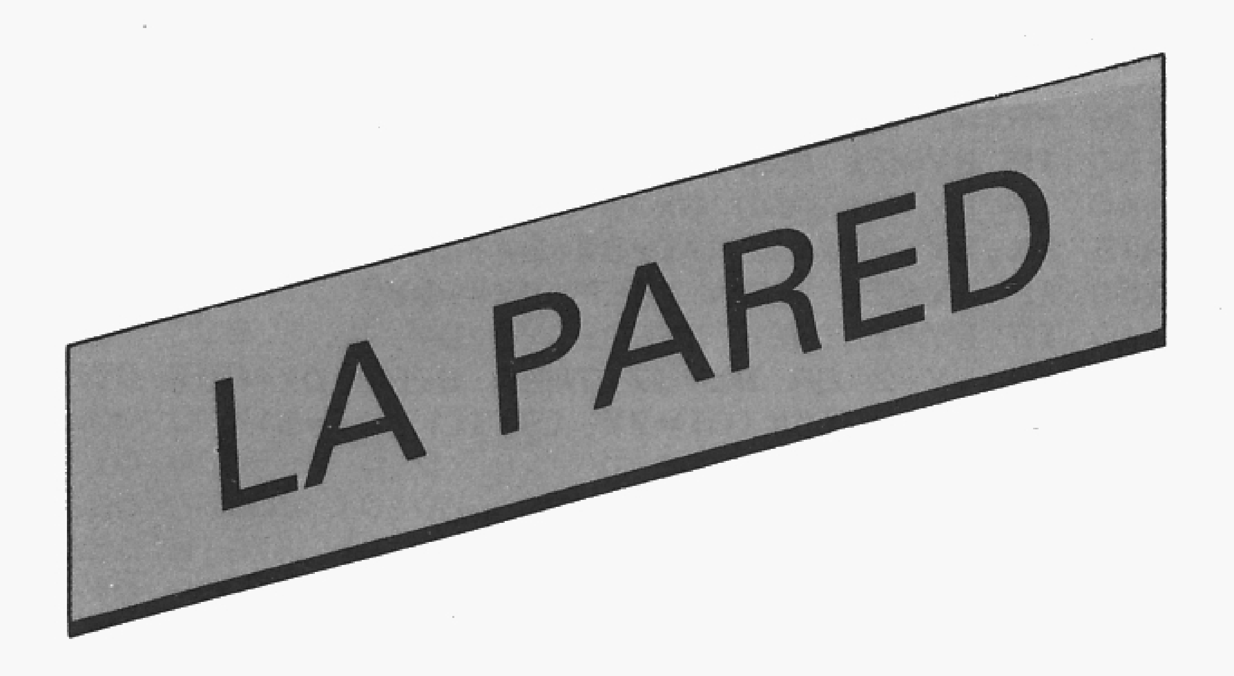
```
130 A$=RIGHT$(A$,1)+LEFT$(A$,38)
140 B$=RIGHT$(B$,38)+LEFT$(B$,1)
150 LOCATE 0,X1,0
160 PRINT A$;
170 LOCATE 0,X2,0
180 PRINT B$;
190 LOCATE 0,X3,0
200 PRINT A$;
210 LOCATE 0,X4,0
220 PRINT B$;
230 D$=INKEY$
240 PY=PY+(STICK(0)=1)-(STICK(0)=5)
250 IF PY>10 THEN PY=10
260 IF PY=2 THEN 370
270 C=VPEEK(PX+PY*40+1)
280 IF C<>32 AND C<>128 THEN 550
290 LOCATE PX,YP,0
300 PRINT N$;
310 LOCATE PX,PY,0
320 PRINT P$;
330 YP=PY
340 T=T+1
350 IF T>500 THEN 660
360 GOTO 110
370 LOCATE PX,YP,0
380 PRINT N$;
390 LOCATE PX,PY,0
400 PRINT P$;
410 BEEP
420 FOR I=1 TO 200
430 NEXT I
440 LOCATE PX,PY,0
450 PRINT N$;
460 PY=10
470 YP=PY
480 S=S+1
490 LOCATE 0,0,0
500 PRINT "PUNTUACION :";S;
510 LOCATE 19,0
520 PRINT "RECORD :";R;
530 GOSUB 1040
540 GOTO 110
```

```
550 NP=NP-1
560 LOCATE PX,YP,0
570 PRINT N$;
580 LOCATE PX,PY,0
590 PRINT CHR$(128);
600 GOSUB 1090
610 IF NP=0 THEN 660
620 PY=10
630 YP=PY
640 GOSUB 1040
650 GOTO 110
660 CLS
670 IF S>R THEN R=S
680 IF T<500 THEN 710
690 LOCATE 10,8,0
700 PRINT "** TIEMPO TRANSCURRIDO **";
710 LOCATE 10,12,0
720 PRINT "PUNTUACION: ";S;
730 LOCATE 10,16,0
740 PRINT "RECORD :";R;
750 LOCATE 10,20
760 PRINT "OTRA ?";
770 IF INKEY$<>"" THEN 770
780 D$=INKEY$
790 IF D$="" THEN 780
800 IF D$<>"N" AND D$<>"n" THEN 100
810 CLS
820 LOCATE 0,0,1
830 END
840 CLS
850 COLOR 1,11
860 P$=CHR$(128)
870 N$=CHR$(32)
880 S=0
890 NP=5
900 PX=19
910 PY=10
920 YP=PY
930 X1=4
940 X2=5
950 X3=7
960 X4=8
```

```
970 T=0
980 RETURN
990 FOR I=1 TO 39
1000 READ A
1010 A$=A$+CHR$(A)
1020 NEXT I
1030 B$=A$
1040 X=RND(1)*35+2
1050 A$=RIGHT$(A$,X)+LEFT$(A$,39-X)
1060 RETURN
1070 DATA 32,129,130,32,32,129,130,32,32
,32,129,130,32,32,32,32,32,129,130,32
1080 DATA 32,32,129,130,32,32,32,129,130
,32,32,129,130,32,32,129,130,32,32
1090 PLAY "T100O3G16O4C2O3G8O4C16E2T150C
8E16G8F+16F8D+16E8C16O3A8G16O4C4C32E16
1100 FOR I=1 TO 4000
1110 NEXT I
1120 RETURN
1130 FOR I=0 TO 23
1140 READ A
1150 VPOKE 3072+I,A*4
1160 NEXT I
1170 RETURN
1180 DATA 12,45,63,30,30,63,45,0
1190 DATA 7,15,31,31,18,16,12,0
1200 DATA 56,60,62,62,18,2,12,0
```







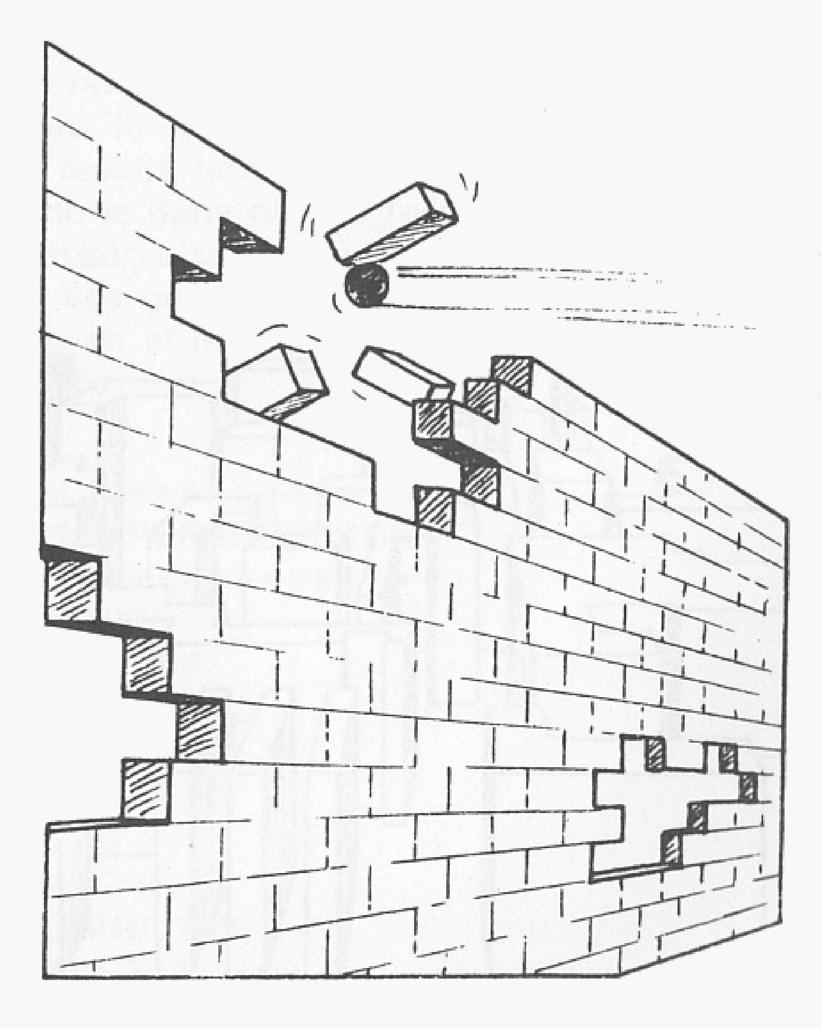
He aquí un juego que se acostumbra a ver en los cafés. Gracias a su ordenador, ahora podrá jugar con él sin gastar ni un céntimo. El objetivo del juego es sencillo: intentar destruir una pared de ladrillos con ayuda de una pelota que debe ser devuelta con su raqueta. Cada ladrillo roto le proporciona un punto. Tan pronto todo el muro está destruido completamente, aparece otro de nuevo. Dispone de diez pelotas para intentar conseguir la máxima puntuación. Para desplazar la raqueta utilice las teclas de control del cursor.

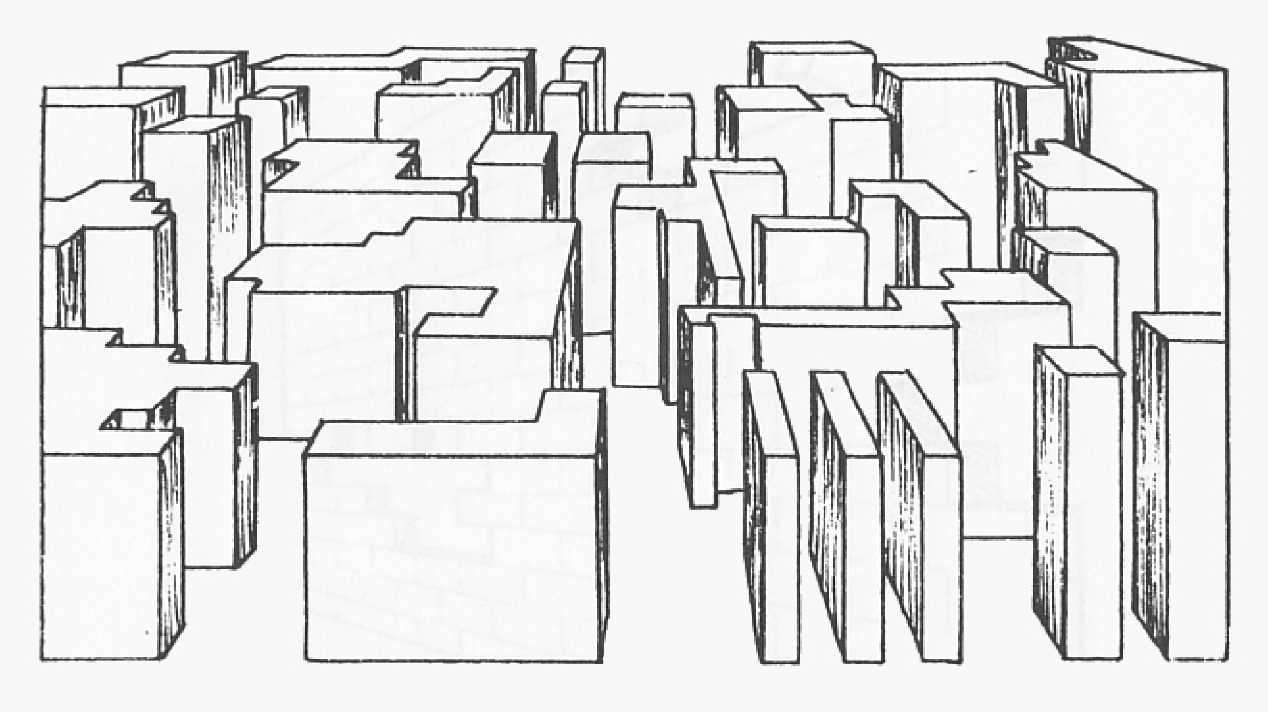
```
10 REM ************
20 REM * LA PARED *
30 REM ************
40 KEY OFF
50 GOSUB 600
60 LOCATE BX,BY,0
70 PRINT N$;
80 BX=BX+DX
90 BY=BY+DY
100 IF VPEEK(BX+BY*40)<>32 THEN DY=-DY:S
=S+1:BEEP
110 LOCATE BX,BY,0
```

```
120 PRINT B$;
130 IF BY=21 AND ABS(BX-RX-3)>1 THEN 290
140 IF BY=21 AND BX>3 AND BX<36 THEN LOC
ATE BX,BY:PRINT N$;:BX=BX+CH
150 IF BY=21 THEN BEEP:DY=-DY
160 IF BY=1 THEN BEEP:DY=-DY
170 IF BX=2 OR BX=37 THEN BEEP:DX=-DX
180 D=2*((STICK(0)=7)-(STICK(0)=3))
190 IF D<>0 THEN DO=D
200 IF STICK(0)=0 THEN DO=0
210 RX=RX+DO
220 CH=SGN(DO)
230 IF RX<0 THEN RX=0
240 IF RX>33 THEN RX=33
250 LOCATE RX,RY,0
260 PRINT R$;
270 IF S=204 THEN GOSUB 930
280 GOTO 60
290 NB=NB+1
300 IF NB=11 THEN 410
310 LOCATE BX,BY,0
320 PRINT N$;
330 FOR I=1 TO 3
340 BEEP
350 FOR J=1 TO 100
360 NEXT J
370 NEXT I
380 DO=0
390 GOSUB 870
400 GOTO 60
410 LOCATE 13,5,0
420 S=S+404*K
430 PRINT "PUNTUACION :";S;
440 IF S>R1 THEN R1=S
450 LOCATE 13,10,0
460 PRINT "RECORD :";R1;
470 LOCATE 13,15,0
480 PRINT "OTRA ?";
490 NB=0
500 K=0
510 S=0
520 D$=INKEY$
```

```
530 IF D$<>"" THEN 520
540 D$=INKEY$
550 IF D$="" THEN 540
560 IF D$<>"N" AND D$<>"n" THEN 50
570 LOCATE 0,0,1
580 CLS
590 END
600 CLS
610 SCREEN 0,0
620 GOSUB 1010
630 WIDTH 40
640 COLOR 15,13
650 DEFINT A-Z
660 BR$=CHR$(128)
670 N$=CHR$(32)
680 R$=N$+N$+CHR$(223)+CHR$(223)+CHR$(22
3)+N$+N$
690 B$=CHR$(249)
700 FOR BX=1 TO 38
710 LOCATE BX,0,0
720 PRINT CHR$(219);
730 NEXT BX
740 FOR BY=1 TO 21
750 LOCATE 1,BY,0
760 PRINT CHR$(219);
770 LOCATE 38,BY,0
780 PRINT CHR$(219);
790 NEXT BY
800 FOR BY=2 TO 7
810 FOR BX=3 TO 36
820 LOCATE BX,BY,0
830 PRINT BR$;
840 NEXT BX
850 NEXT BY
860 RX=16
870 RY=22
880 BY=21
890 BX=INT(RND(1)*34)+3
900 DY=-1
910 DX=(INT(RND(1)*2)-.5)*2
920 RETURN
930 K=K+1
```

```
940 S=0
950 LOCATE BX,BY,0
960 PRINT N$;
970 LOCATE RX+2,RY,0
980 PRINT N$,N$,N$;
990 GOSUB 700
1000 RETURN
1010 RESTORE
1020 FOR I=0 TO 7
1030 READ A
1040 VPOKE 3072+I,A*4
1050 NEXT I
1060 RETURN
1070 DATA 63,33,33,33,33,33,33,63
```







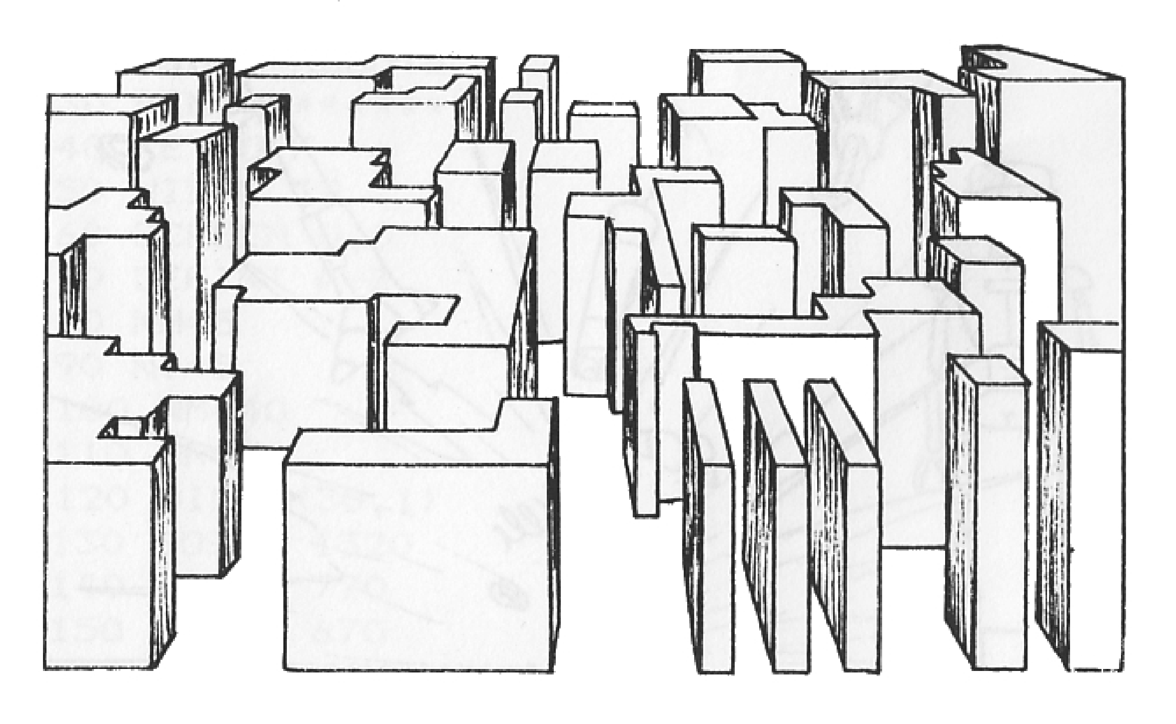
El ladrón se ha escapado, llevándose el botín. (Está representado por un as de pique). Se esconde en la ciudad, y Vd. tiene cincuenta minutos para encontrarlo y detenerlo. Atención, ¡no se precipite! Efectivamente, si se echa sobre él sin pensar, tiene todas las posibilidades de que se le escape de entre los dedos. La mejor manera de cogerlo es alcanzarlo de lado. (Es eficaz si actúa de repente, a condición de que no falle). Si no se siente demasiado seguro de sí mismo, atáquele de cara, lo cual es más fácil pero mucho menos eficaz, ya que no resulta tan discreto. *Otro consejo:* no intente perseguirlo; no le daría ningún resultado, pues es más rápido que usted. Observe sus movimientos, como si fuese un detective. Cuando vea que da vuelta, acérquese sin hacer ruido y sorpréndalo en el momento preciso. Pero, recuerde, ¡el tiempo va pasando!

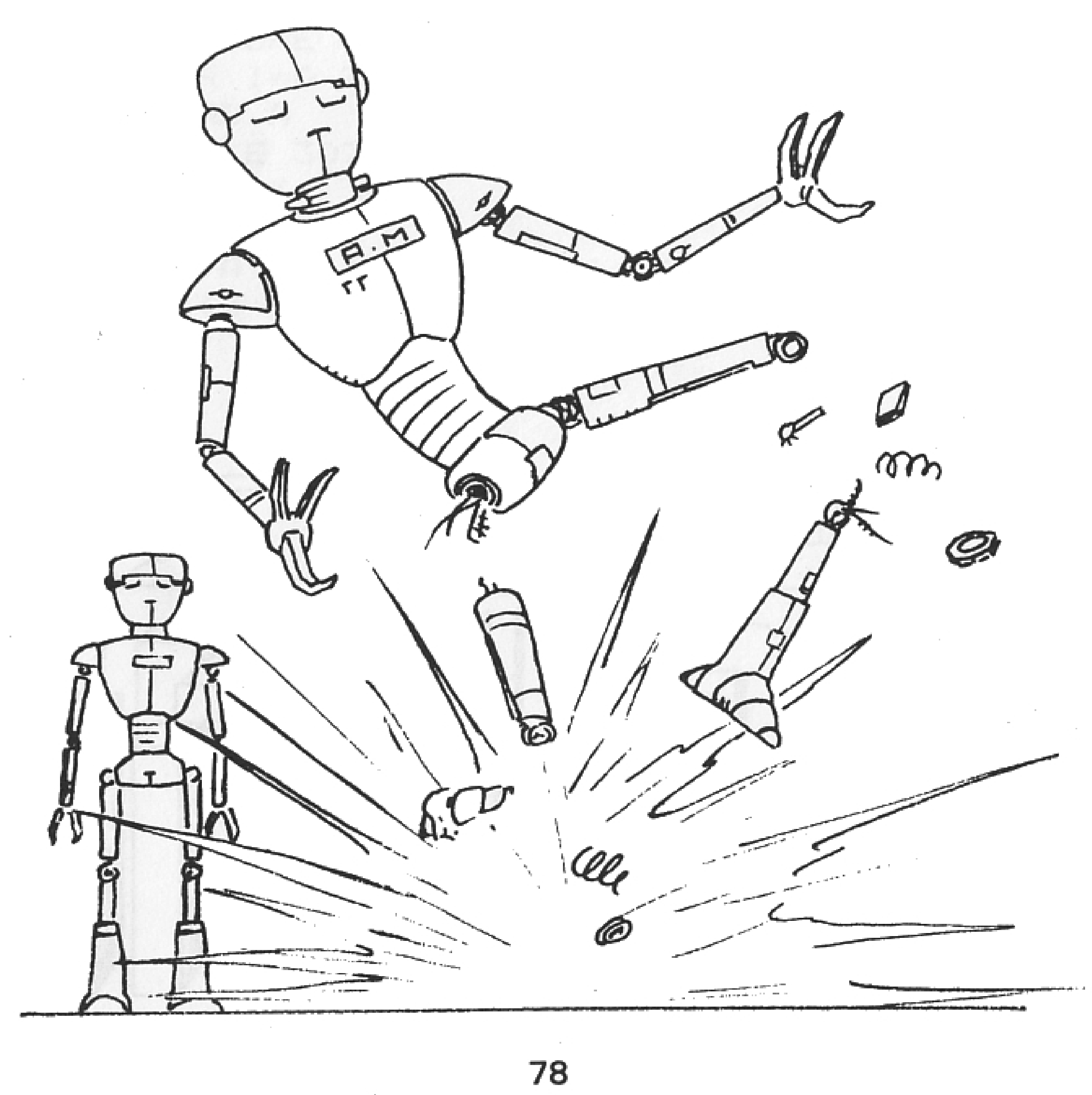
```
10 REM *************
20 REM *PERSECUCION*
30 REM *************
40 KEY OFF
50 WIDTH 39
60 GOSUB 1060
70 S=0
80 N=32
90 V=128
100 P=129
110 GOSUB 670
120 ST=STICK(0)
```

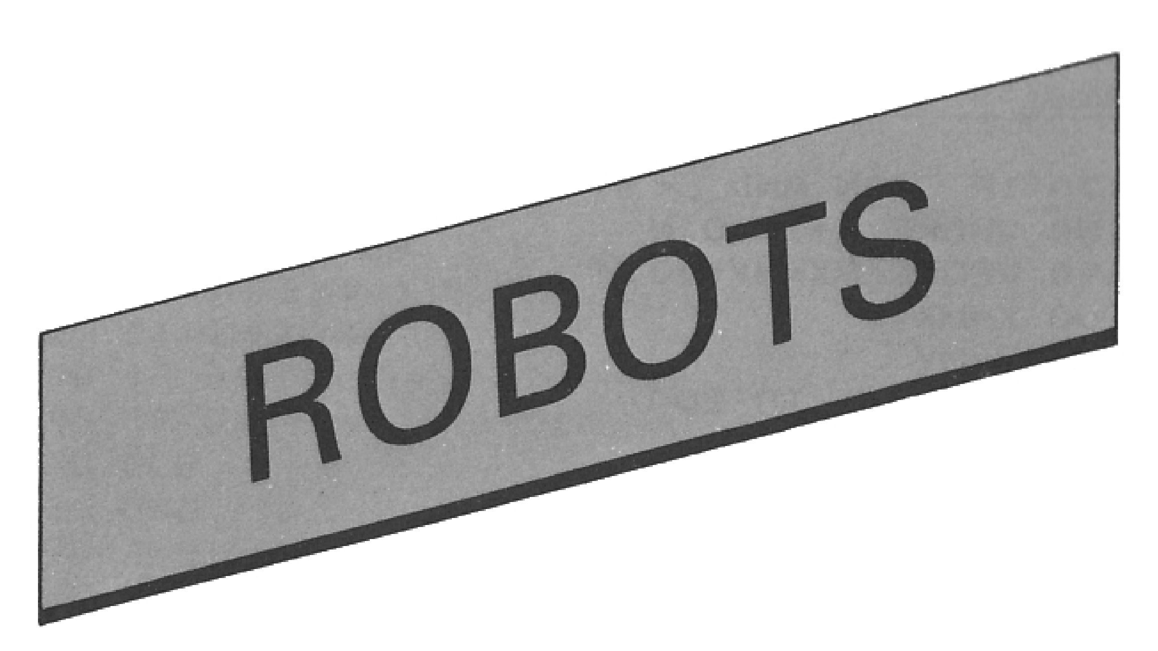
```
130 DX=(ST=7)-(ST=3)
140 DY=(ST=1)-(ST=5)
150 Z=Z-.2
160 LOCATE 0,24,0
170 PRINT "TIEMPO:";INT(Z+1);
180 IF Z<0 THEN 370
190 PX=PX+DX
200 PY=PY+DY
210 C=VPEEK(PX+PY*40)
220 IF C=V THEN 1010
230 IF C<>32 THEN PX=XP:PY=YP
240 VPOKE XP+YP*40,N
250 VPOKE PX+PY*40,P
260 YP=PY
270 XP=PX
280 VX=VX+CX
290 VY=VY+CY
300 IF VPEEK(VX+VY*40)<>N THEN GOSUB 510
310 IF VPEEK(VX+VY*40)<>N THEN 280
320 VPOKE XV+YV*40,N
330 VPOKE VX+VY*40,V
340 XV=VX
350 YV=VY
360 GOTO 120
370 IF INKEY$<>"" THEN 370
380 FOR I=1 TO 1000
390 NEXT I
400 LOCATE 10,6
410 PRINT "TIEMPO TRANSCURRIDO ";
420 LOCATE 10,10
430 PRINT "PUNTUACION :";S;
440 LOCATE 10,18,1
450 PRINT "OTRA ?";
460 D$=INKEY$
470 IF D$="" THEN 460
480 IF D$<>"N" AND D$<>"n" THEN RUN
490 CLS
500 END
510 DT=DT+1
520 GOSUB 620
530 IF VPEEK(XV+CX+(YV+CY)*40)=N THEN VX
=XV+CX:VY=YV+CY:RETURN
```

```
540 DT=DT-2
550 GOSUB 620
560 IF VPEEK(XV+CX+(YV+CY)*40)=N THEN VX
=XV+CX:VY=YV+CY:RETURN
570 DT=DT-1
580 GOSUB 620
590 VX=XV+CX
600 VY=YV+CY
610 RETURN
620 IF DT>4 THEN DT=DT-4
630 IF DT<1 THEN DT=DT+4
640 CX=(DT=1)-(DT=3)
650 CY=(DT=2)-(DT=4)
660 RETURN
670 CLS
680 FOR VX=0 TO 39
690 VPOKE VX+40,219
700 VPOKE VX+880,219
710 NEXT VX
720 FOR VY=2 TO 22
730 VPOKE VY*40,219
740 VPOKE VY*40+39,219
750 NEXT VY
760 FOR VX=1 TO 150
770 GOSUB 970
780 VPOKE PX+PY*40,219
790 NEXT VX
800 GOSUB 970
810 VX=PX
820 VY=PY
830 VPOKE VX+VY*40,V
840 XV=VX
850 YV=VY
860 GOSUB 970
870 VPOKE PX+PY*40,P
880 XP=PX
890 YP=PY
900 Z=50
910 CX=0
920 CY=0
930 DX=0
940 DY=0
```

```
950 DT=0
960 RETURN
970 PX=INT(RND(1)*37)+1
980 PY=INT(RND(1)*21)+2
990 IF VPEEK(PX+PY*40)<>N THEN 970
1000 RETURN
1010 FOR I=1 TO 5
1020 PLAY "DC"
1030 NEXT I
1040 S=S+1
1050 GOTO 110
1060 SCREEN 0,0
1070 COLOR 6,14
1080 CLS
1090 DEFINT A-Y
1100 PX=RND(-TIME)
1110 RESTORE
1120 FOR I=0 TO 15
1130 READ A
1140 VPOKE 3072+I,A*4
1150 NEXT I
1160 RETURN
1170 DATA 30,63,45,63,30,30,30,0
1180 DATA 28,28,8,62,8,8,20,20
```




A.M



Está solo, abandonado sobre un planeta defendido por robots asesinos. El suelo está sembrado de minas las cuales debe evitar necesariamente. Las minas se hallan representadas sobre la pantalla mediante unas "X". Al comienzo del juego, cinco robots se hallan sobre el terreno. Sin perder un segundo, se precipitan hacia usted, siguiendo siempre el camino más corto. Por suerte, los robots son ciegos y no ven las minas que se hallan entre ellos y usted, lo cual le permite eliminarlos, siempre que se desplace del modo adecuado. Para ello debe usar las teclas del cursor.

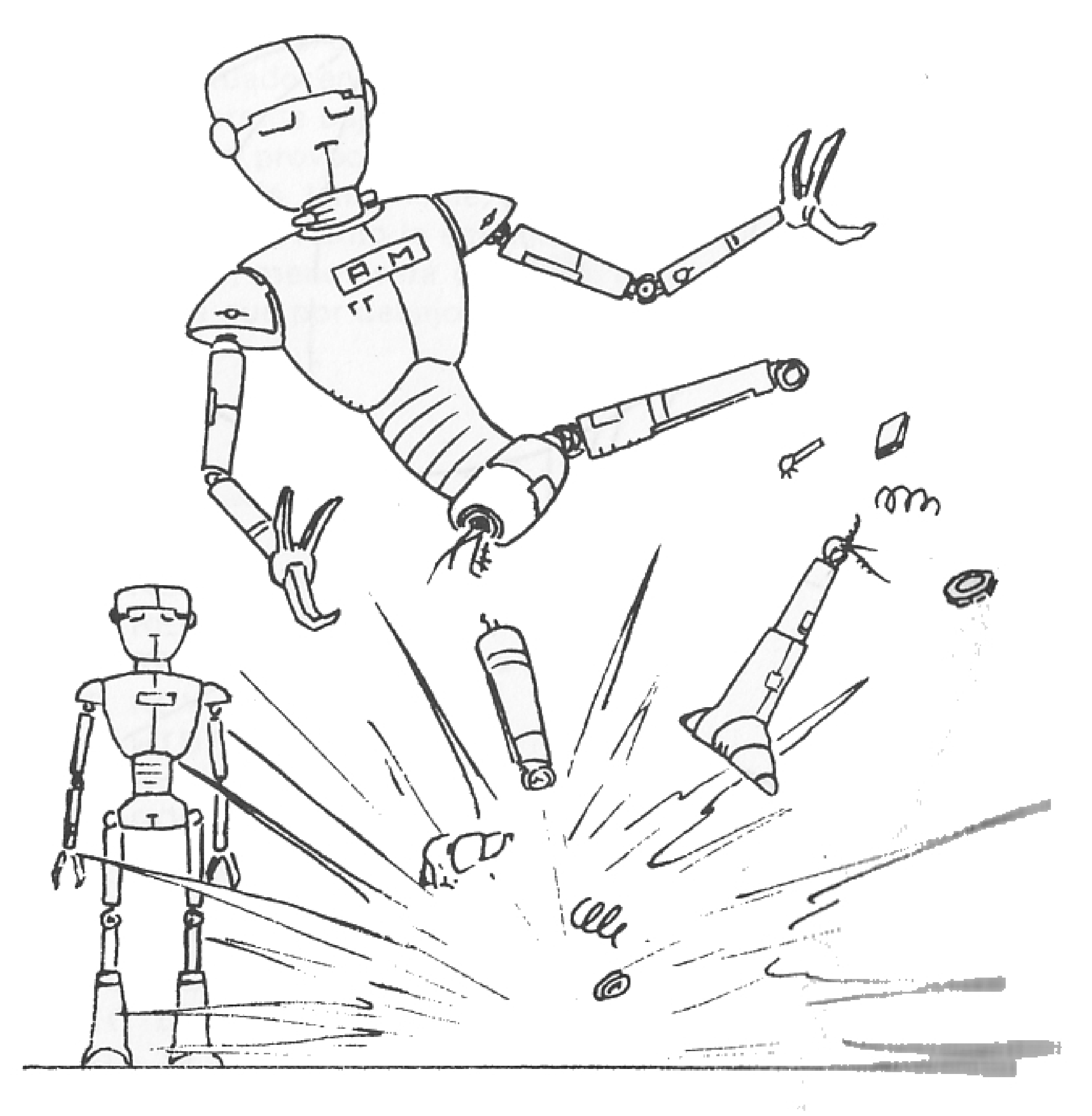
Cuando haya eliminado a todos los robots, el juego continúa con un robot suplementario. Si salta sobre una de las minas o le mata uno de los robots, aún no se ha perdido todo. En realidad tiene cinco vidas.

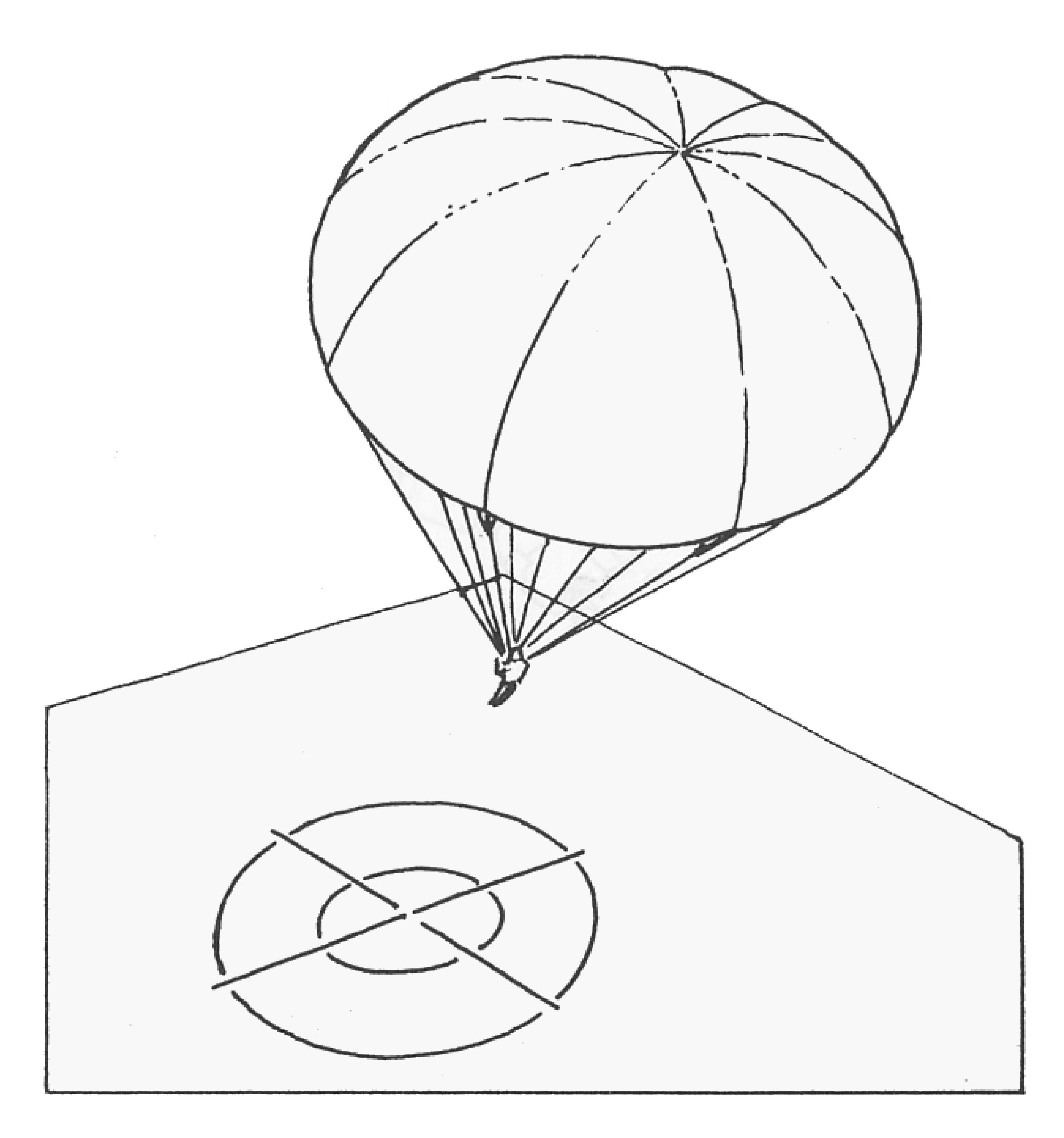
```
10 REM **********
20 REM * ROBOTS *
30 REM **********
40 KEY OFF
50 WIDTH 39
60 SCREEN 0,0
70 DEFINT A-Z
80 NH=5
90 N1=5
100 NM=40
110 NR=N1
120 DIM R(30,1)
130 GOSUB 1320
140 GOSUB 770
150 GOSUB 670
160 C=VPEEK(HX+HY*40)
```

```
170 IF C<>N AND C<>H THEN 440
180 VPOKE X+Y*40,N
190 VPOKE HX+HY*40,H
200 X=HX
210 Y=HY
220 FOR J=1 TO 200
230 NEXT J
240 T=0
250 FOR I=1 TO NR
260 IF R(I,0)=0 THEN 370
270 T=1
280 RX=R(I,0)+SGN(HX-R(I,0))
290 RY=R(I,1)+SGN(HY-R(I,1))
300 C=VPEEK(RX+RY*40)
310 IF C=M OR C=R THEN S=S+1:VPOKE R(I,0
)+40*R(I,1),N:R(I,0)=0:GOTO 370
320 IF C=H THEN 440
330 VPOKE R(I,0)+40*R(I,1),N
340 VPOKE RX+40*RY,R
350 R(I,0)=RX
360 R(I,1)=RY
370 NEXT I
380 IF T=0 THEN 400
390 GOTO 150
400 S=S+10
410 IF INKEY$<>"" THEN 410
420 IF NR<30 THEN NR=NR+1
430 GOTO 140
440 NH=NH-1
450 VPOKE X+40*Y,N
460 VPOKE HX+40*HY,H
470 A1$="V12D2D3D8D2F3E8E3D8D3C+8D2"
480 A2$="F2F3F8F2A3G8G3F8F3E8F2"
490 A3$="O3A2B-3B-8A2B-3B-8A3A8B-3B-8A2"
500 PLAY A1$,A2$,A3$
510 FOR I=1 TO 5000
520 NEXT I
530 IF INKEY$<>"" THEN 530
540 IF NH>0 THEN NR=N1:GOTO 140
550 CLS
560 SCREEN 0,0
570 LOCATE 9,10
```

```
580 PRINT "PUNTUACION :";S;
590 LOCATE 9,20
600 PRINT "OTRA ?";
610 IF INKEY$<>"" THEN 610
620 D$=INKEY$
630 IF D$="" THEN 620
640 IF D$<>"N" AND D$<>"n" THEN RUN
650 CLS
660 END
670 J=STICK(0)
680 IF J=1 THEN HY=HY-1
690 IF J=2 THEN HY=HY-1:HX=HX+1
700 IF J=3 THEN HX=HX+1
710 IF J=4 THEN HX=HX+1:HY=HY+1
720 IF J=5 THEN HY=HY+1
730 IF J=6 THEN HY=HY+1:HX=HX-1
740 IF J=7 THEN HX=HX-1
750 IF J=8 THEN HX=HX-1:HY=HY-1
760 RETURN
770 CLS
780 COLOR 1,13
790 PRINT "PUNTUACION ";S
800 IF NH=1 THEN 850
810 FOR HX=1 TO NH-1
820 LOCATE 19+HX,24,0
830 PRINT H$;
840 NEXT HX
850 FOR HX=0 TO 39
860 VPOKE HX+40,M
870 VPOKE HX+880,M
880 NEXT HX
890 FOR HY=2 TO 22
900 VPOKE HY*40,M
910 VPOKE HY*40+39,M
920 NEXT HY
930 FOR I=1 TO NM
940 HX=INT(RND(1)*37)+1
950 HY=INT(RND(1)*21)+2
960 VPOKE HX+40*HY,M
970 NEXT I
980 FOR I=1 TO NR
990 R(I,0)=INT(RND(1)*37)+1
```

```
1000 R(I,1)=INT(RND(1)*21)+2
1010 IF VPEEK(R(I,0)+40*R(I,1))<>32 THEN
 990
1020 VPOKE R(I,0)+40*R(I,1),R
1030 NEXT I
1040 HX=INT(RND(1)*37)+1
1050 HY=INT(RND(1)*21)+2
1060 IF VPEEK(HX+HY*40)<>32 THEN 1040
1070 IF VPEEK(HX+HY*40-41)<>32 THEN 1040
1080 IF VPEEK(HX+HY*40-40)<>32 THEN 1040
1090 IF VPEEK(HX+HY*40-39)<>32 THEN 1040
1100 IF VPEEK(HX+HY*40-1)<>32 THEN 1040
1110 IF VPEEK(HX+HY*40+1)<>32 THEN 1040
1120 IF VPEEK(HX+HY*40+39)<>32 THEN 1040
1130 IF VPEEK(HX+HY*40+40)<>32 THEN 1040
1140 IF VPEEK(HX+HY*40+41)<>32 THEN 1040
1150 X=HX
1160 Y=HY
1170 FOR I=1 TO 5
1180 VPOKE HX+HY*40,H
1190 BEEP
1200 FOR J=1 TO 50
1210 NEXT J
1220 VPOKE HX+HY*40,H
1230 FOR J=1 TO 50
1240 NEXT J
1250 NEXT I
1260 RESTORE
1270 FOR I=0 TO 15
1280 READ A
1290 VPOKE 3072+I,A*4
1300 NEXT I
1310 RETURN
1320 H=129
1330 H$=CHR$(129)
1340 M=219
1350 R=128
1360 N=32
1370 RETURN
1380 DATA 30,63,45,63,63,18,30,30
1390 DATA 28,28,8,62,8,8,20,20
```


A·M



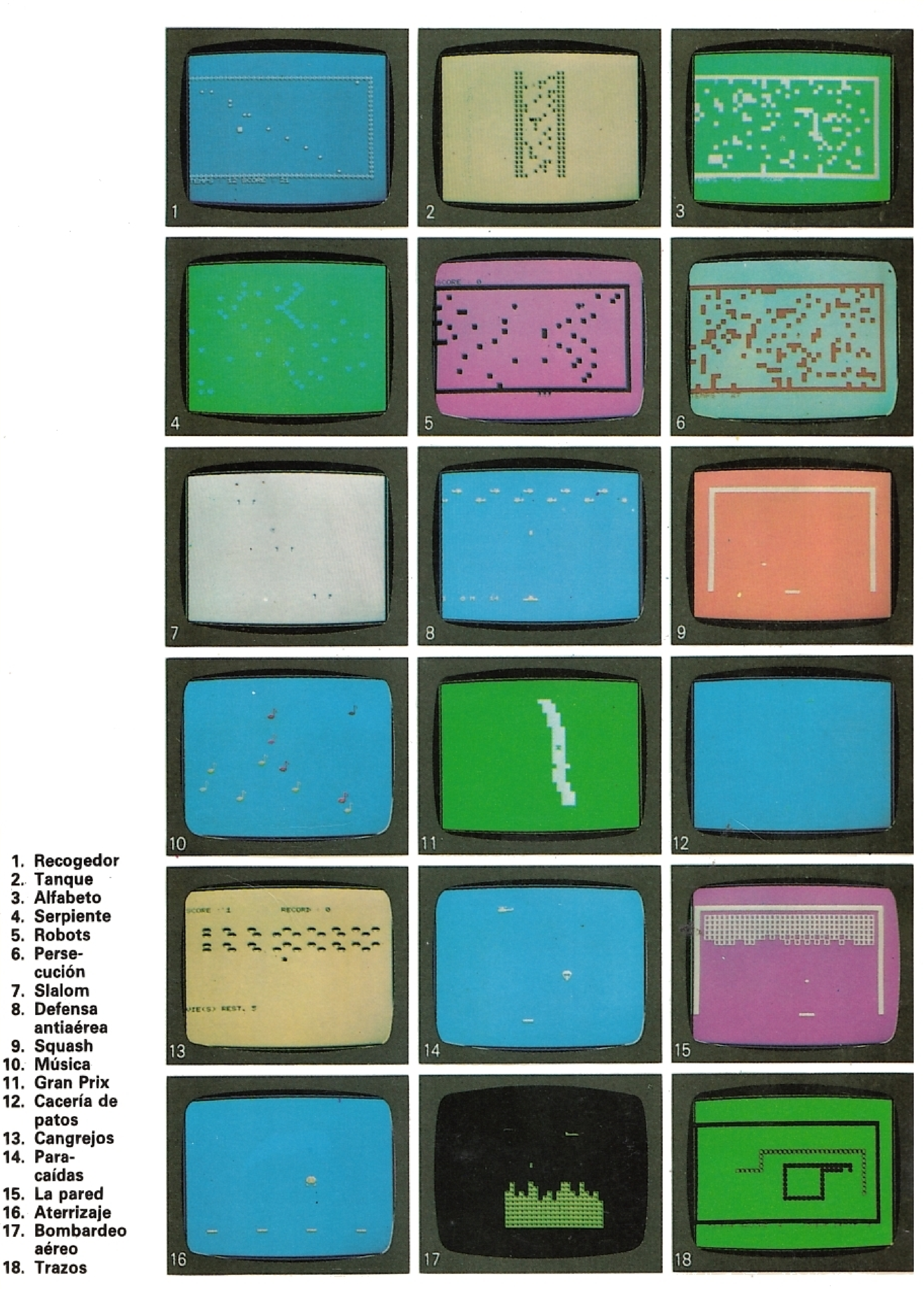


Saltando de un helicóptero en vuelo, intente alcanzar el blanco situado en el suelo. Una primera presión sobre una tecla le permite bajar verticalmente en caída libre. Una segunda presión provoca la abertura del paracaídas. El descenso continúa más lentamente, y con un ángulo de 45 grados, puesto que el viento le empuja. Cuando más espere a abrir el paracaídas, menos será desviado. Pero, no aguarde excesivamente ya que por debajo de los cien metros, el paracaídas no se abrirá.

```
10 REM *************
20 REM * PARACAIDAS *
30 REM *************
40 KEY OFF
50 WIDTH 39
60 S=0
70 GOSUB 510
80 H=H-1
90 IF H=-1 THEN LOCATE 0,0:PRINT M$;:HH=
36:H=36
100 LOCATE H,0:PRINT H$;
110 D$=INKEY$
120 IF D$="" THEN GOTO 170
```

```
130 IF PV>16 THEN GOTO 170
140 IF SP=1 THEN LET OP=1:GOTO 160
150 SP=1
160 IF OP=0 THEN PV=1:PH=H+1.5
170 IF SP=0 THEN GOTO 340
180 IF OP=0 THEN PV=PV+1
190 IF OP=1 THEN PV=PV+.5:PH=PH-.5
200 IF PV>21 OR PH<0 THEN GOTO 360
210 IF OP=1 THEN GOTO 280
220 LOCATE HO,V:PRINT N$+N$;
230 V=PV
240 HO=PH
250 LOCATE HO,V:PRINT C$;
260 FOR I=1 TO 40:NEXT I
270 GOTO 80
280 LOCATE HO,V:PRINT N$+N$;
290 LOCATE HO,V+1:PRINT N$;
300 V=PV
310 HO=PH
320 LOCATE HO,V:PRINT P$;
330 LOCATE HO,V+1:PRINT Q$;
340 FOR I=1 TO 60:NEXT I
350 GOTO 80
360 IF ABS((HO-.5)-PA)>1 THEN GOTO 420
370 FOR I=1 TO 500
380 NEXT I
390 S=S+1
400 GOSUB 510
410 GOTO 80
420 CLS
430 LOCATE 10,10:PRINT "PUNTUACION :";S;
440 LOCATE 10,14:PRINT "OTRA ?";
450 D$=INKEY$
460 IF D$="" THEN GOTO 450
470 IF D$<>"N" AND D$<>"n" THEN RUN
480 CLS
490 LOCATE 0,0,1
500 END
510 SCREEN 0,0
520 COLOR 15,5
530 RESTORE
540 FOR I=0 TO 79
```

```
550 READ A
560 VPOKE 3072+I,A*4
570 NEXT I
580 N$=CHR$(32)
590 M$=N$+N$+N$
600 H$=CHR$(128)+CHR$(129)+CHR$(130)+N$
610 C$=CHR$(131)+CHR$(132)
620 P$=CHR$(133)+CHR$(134)+N$
630 Q$=CHR$(135)+CHR$(136)
640 A$=CHR$(137)+CHR$(137)
650 H=36
660 PA=INT(RND(1)*28)+1
670 LOCATE PA,24,0:PRINT A$;
680 SP=0
690 OP=0
700 PH=0
710 PV=0
720 CC=2
730 V=0
740 HO=0
750 RETURN
760 DATA 63,1,15,31,31,63,63,31
770 DATA 62,0,32,56,63,63,60,48
780 DATA 0,0,3,3,63,62,0,0
790 DATA 2,2,3,3,15,9,11,3
800 DATA 32,32,32,32,56,8,40,32
810 DATA 7,31,31,63,63,32,32,16
820 DATA 48,60,60,62,62,2,2,4
830 DATA 19,11,9,15,3,3,2,2
840 DATA 36,40,8,56,32,32,32,32
850 DATA 63,63,63,0,0,0,0,0
```



1. **Recogedor**
2. **Tanque**
3. **Alfabeto**
4. **Serpiente**
5. **Robots**
6. **Persecución**
7. **Slalom**
8. **Defensa antiaérea**
9. **Squash**
10. **Música**
11. **Gran Prix**
12. **Cacería de patos**
13. **Cangrejos**
14. **Paracaídas**
15. **La pared**
16. **Aterrizaje**
17. **Bombardeo aéreo**
18. **Trazos**